White paper
Cisco public

Tuning Guidelines for Cisco UCS Virtual Interface Cards

May 11, 2021

Contents

The goal of this document is to help administrators optimize network I/O performance both in standalone Cisco Unified Computing System™ (Cisco UCS®) servers and in configurations operating under Cisco UCS Manager. Note that although this document covers I/O performance optimization for the Cisco UCS virtual interface card (VIC), factors such as CPU, BIOS, memory, operating system, and kernel can contribute to overall I/O workload performance. For more specific performance optimization recommendations, please refer to Cisco® BIOS best practices tuning guides and Cisco Validated Designs for specific workloads.

Introduction

The Cisco UCS virtual interface card, or VIC, is a converged network adapter (CNA) designed for Cisco UCS blade and rack servers. The Cisco VIC is a stateless hardware device that is software programmable, providing management, data, and storage connectivity for Cisco UCS servers. Installed as a part of Cisco UCS or a standalone environment, the VIC is used to create PCI Express (PCIe) standards-compliant virtual interfaces: both virtual network interfaces (vNICs) and virtual host bus adapters (vHBAs). Indistinguishable from hardware NICs and HBAs, these interfaces can be dynamically defined and configured at the time of server provisioning.

Unless otherwise noted, this document is applicable to all past and present Cisco UCS VICs—for both blade and rack server platforms—operating under either Cisco UCS Manager or Cisco Integrated Management Controller (IMC).

Audience

The target audiences for this document are systems architects, system and server administrators, and any other technical staff who are responsible for managing Cisco UCS servers. Although this document is intended to appeal to the widest possible audience, the document assumes that the reader has an understanding of Cisco UCS hardware, terminology, and configuration.

Cisco UCS VICs and vNICs

The sections that follow describe the Cisco UCS VICs, vNICs, and the capabilities of various configurations.

Cisco UCS VIC overview

The Cisco UCS VIC is available in a variety of models and form factors, allowing it to be supported in both Cisco UCS blade and rack servers (Figure 1). Depending on the generation and model, the adapter includes a PCIe 2.0 or 3.0 interface with either x8 or x16 connectivity and 10-, 20-, 25-, 40-, 50-, and 100-Gbps port speeds. For more specific information about capabilities, speed, operation, and server and network connectivity, refer to the data sheets for individual Cisco UCS VIC and server models.

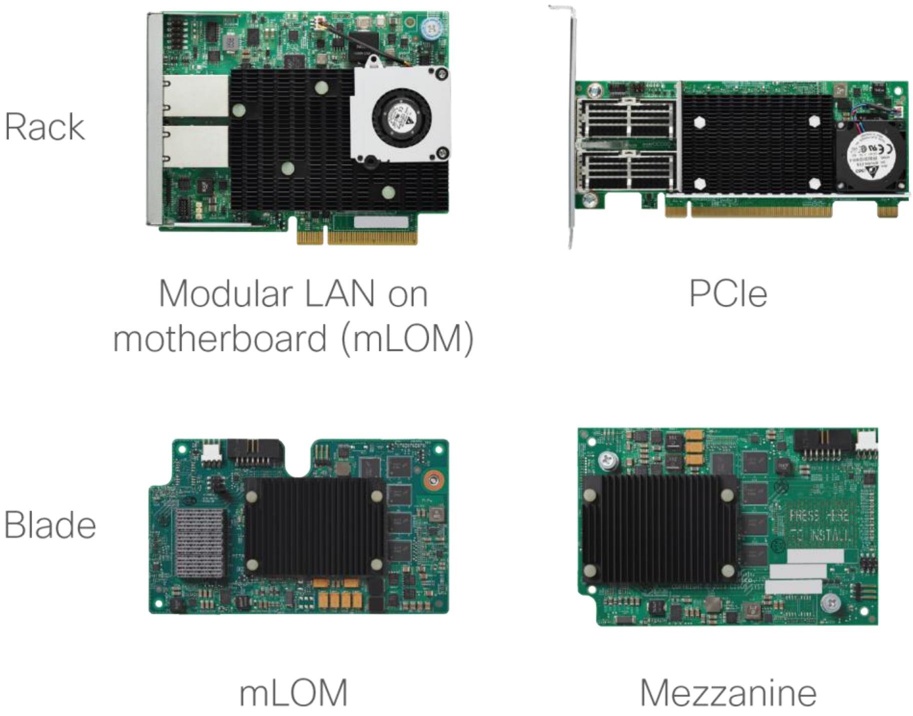


Figure 1. Cisco UCS VIC form factors accommodate both blade and rack Cisco UCS servers

Through unique Cisco® technology and policies, each Cisco UCS VIC provides up to 256 PCIe interfaces (the number depends on the Cisco UCS VIC model). Each virtual interface (vNIC or vHBA) created on the Cisco VIC application-specific integrated circuit (ASIC) is presented to the operating system as a fully standards-compliant PCIe bridge and endpoints. Each vNIC gets its own PCI address, memory space, interrupts, and so forth. The vNIC does not require any special driver and is supported as a part of the standard OS installation package.

Figure 2 provides a logical view of the Cisco UCS VIC, including its dual connections to a redundant pair of Cisco UCS fabric interconnects or LANs. The kernel's standard Ethernet NIC (eNIC) driver allows the OS to recognize the vNICS. Having the most current eNIC driver can help improve network I/O performance, and Cisco recommends that the driver be updated based on the Cisco UCS Manager or Cisco IMC firmware and OS version level. The recommended driver level can be found through the Cisco UCS Hardware and Software Interoperability Matrix Tool (see the link at the end of this document).

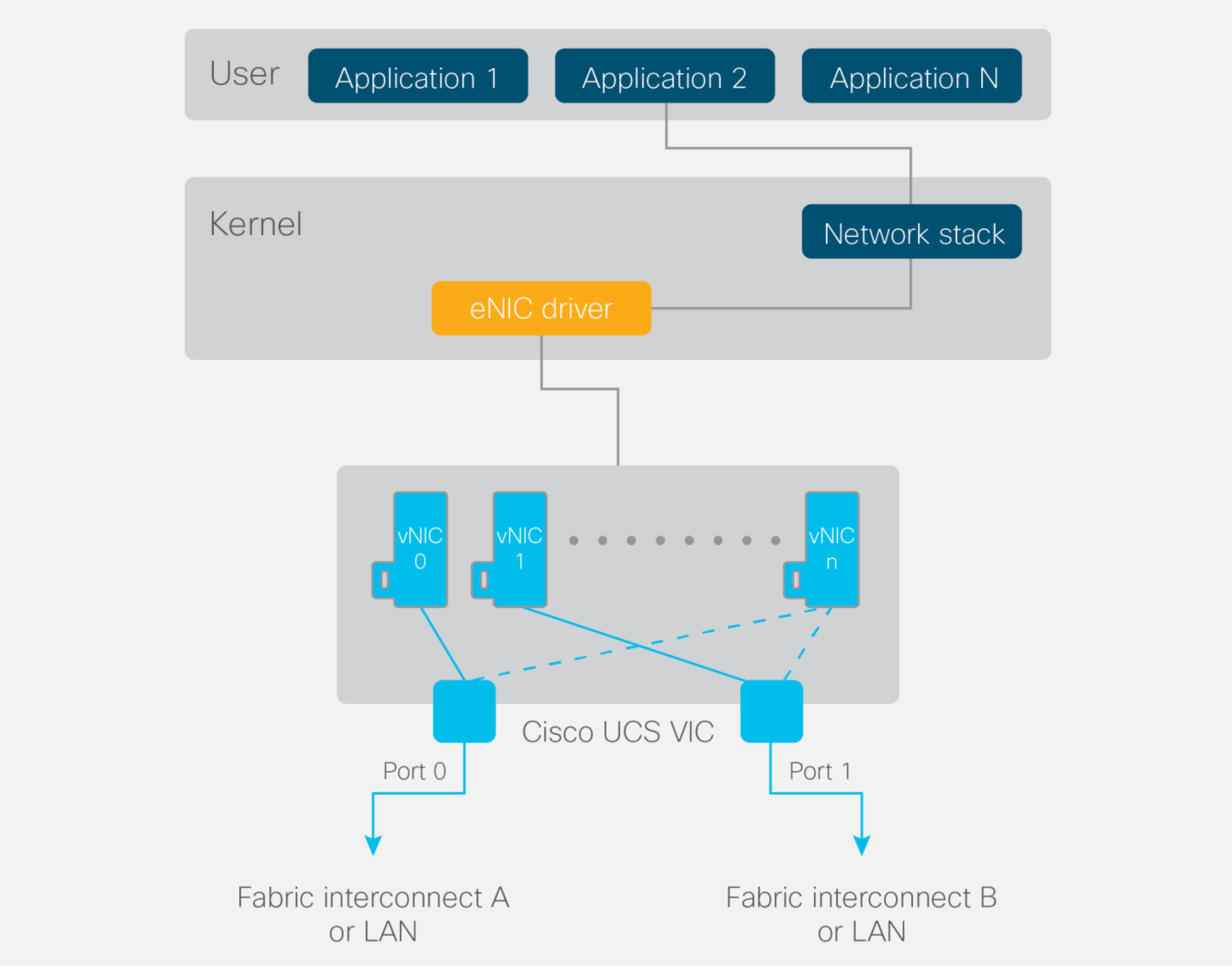


Figure 2. Logical view of the Cisco UCS VIC and its connections

vNIC overview

By default, two vNICs are created on each Cisco UCS VIC, with one bound to each of the adapter's interfaces (ports 0 and 1). The server administrator can create additional vNICs to help segregate the different types of traffic that will be flowing through the adapter. For example, as shown in Figure 3, a server administrator might create four vNICs as follows:

- Two vNICs for management (one for side-A connectivity and one for side-B connectivity)
- Two additional vNICs for data (one for side-A connectivity and one for side-B connectivity)

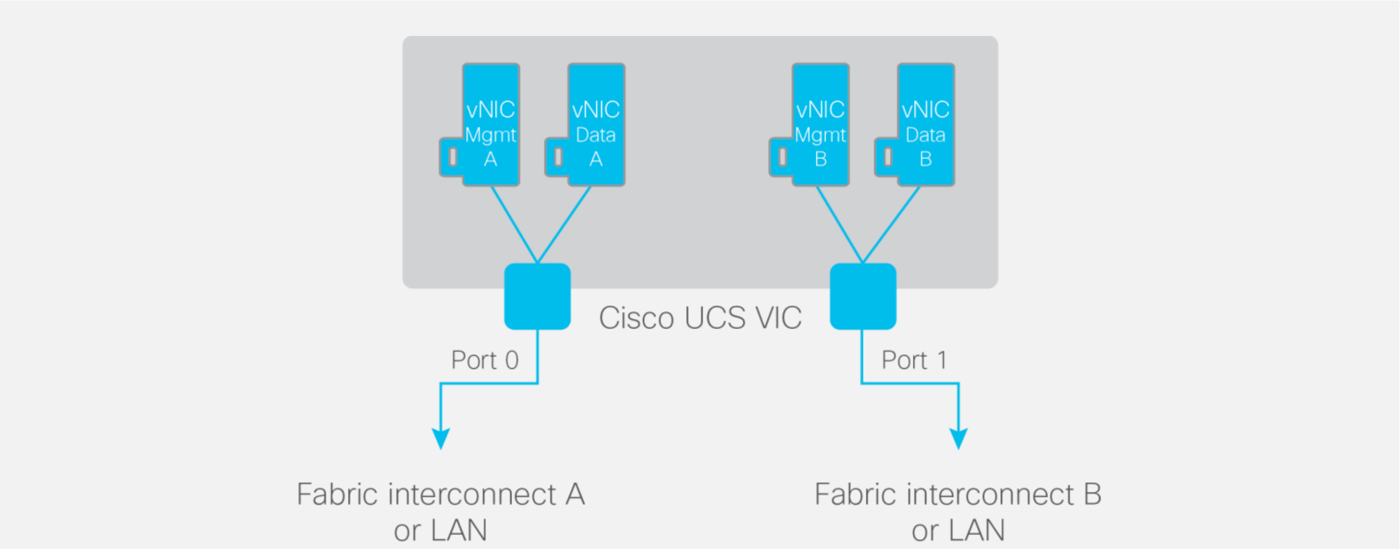


Figure 3. Four vNICs are created for redundant data and management connections to Cisco UCS fabric interconnects or LANs

The total number of supported vNICs is OS specific, and each OS allocates a different number of interrupts to the adapters. For the exact number of supported vNICs, refer to the Configuration Limit Guide on Cisco.com (see the link at the end of this document).

The 1200, 1300 and 1400 series Cisco UCS rack server VICs have two external ports, except for the 4th-generation Cisco UCS VICs 1455 1457, and 1467, which have four external adapter interfaces. On the Cisco UCS VIC 1455, 1457 and 1467, ports (1,2), and ports (3,4), are hardware port-channeled by default. This hardware port-channel is internal to the VIC 1455, 1457, and 1467 hardware. The hardware port-channel cannot be disabled in Cisco UCS Manager mode but when servers are in standalone mode the VIC hardware port-channel can be disabled from the Cisco IMC.

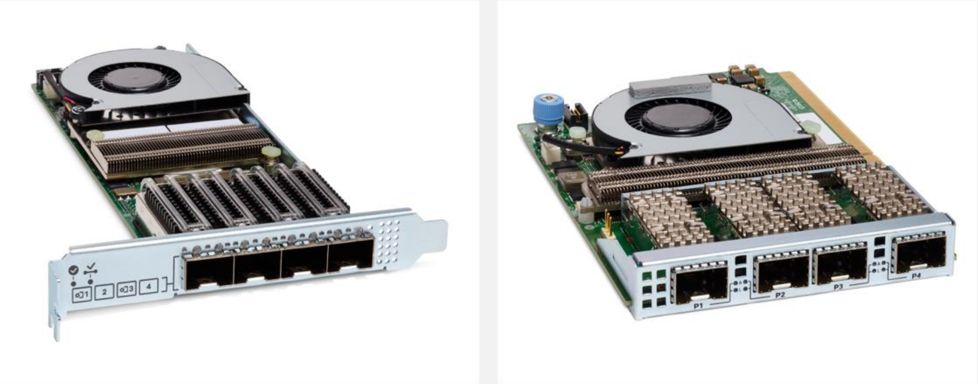


Figure 4. Cisco UCS VIC 1455 and VIC 1457

Understanding port speeds

The port speed that is ultimately presented to the OS varies depending on the network connectivity.

- **For Cisco UCS C-Series Rack Servers**, the port speed is straightforward: it is simply the physical port speed of the PCIe adapter. Exception to this would be the 4-port Cisco UCS VICs 1455, 1457, and 1467 with the default VIC hardware port-channeled enabled, and in this case the port speed seen by the operating system is that of the port-channel (10-, 20-, 25- or 50-Gbps) depending on how many links of the port-channel are up and at what speed. In port-channel mode for VIC 1455, 1457, and 1467, each vNIC on the card is bound to one of the hardware port-channels.
- **For Cisco UCS B-Series Blade Servers**, the calculation is more complicated, because the number of connections from the adapter to the Cisco UCS fabric extender (also known as an I/O module, or IOM) varies depending on the model of both.

Table 1 compares the type of connectivity and the speed presented to the OS based on the Cisco UCS VIC blade adapters in combination with different Cisco UCS I/O module models and a Cisco UCS blade server. For example, when connected to higher-bandwidth Cisco UCS 2208, 2304, and 2408 Fabric Extender I/O modules, the adapter will have multiple 10-Gbps connections to the I/O module. For these cases, the multiple 10-Gbps connections are automatically combined into a hardware port channel (Figure 5)—a process that is transparent to the OS. The Cisco UCS VIC driver will then present the aggregate speed for the vNIC to the OS. The traffic distribution across the port channel is hashed in hardware across the physical links based on Layer 2, Layer 3, and Layer 4 information.

To help define available bandwidth, a flow is defined as a single TCP connection between two servers. As a result, although the OS may show that the vNIC can provide 20 or 40 Gbps of bandwidth (depending on the adapter combination), the maximum throughput for a single flow may be only 10, 25, or 40 Gbps because of the physical port connection. However, if there are multiple flows (for example, multiple TCP connections), then the aggregate throughput for the same vNIC can be 20 Gbps, assuming that the flows is hashed across the two 10-Gbps connections (or 40 Gbps across four 10-Gbps connections). In the case of the Cisco UCS 2304 fabric extender, the server can connect at a true 40 Gbps (depending on the adapter combination), allowing the flow to burst up to the entire 40-Gbps bandwidth of the connection. And with Cisco UCS 2408 fabric extender, the server can allow a single-flow burst of up to 25 Gbps (depending on the adapter combination).

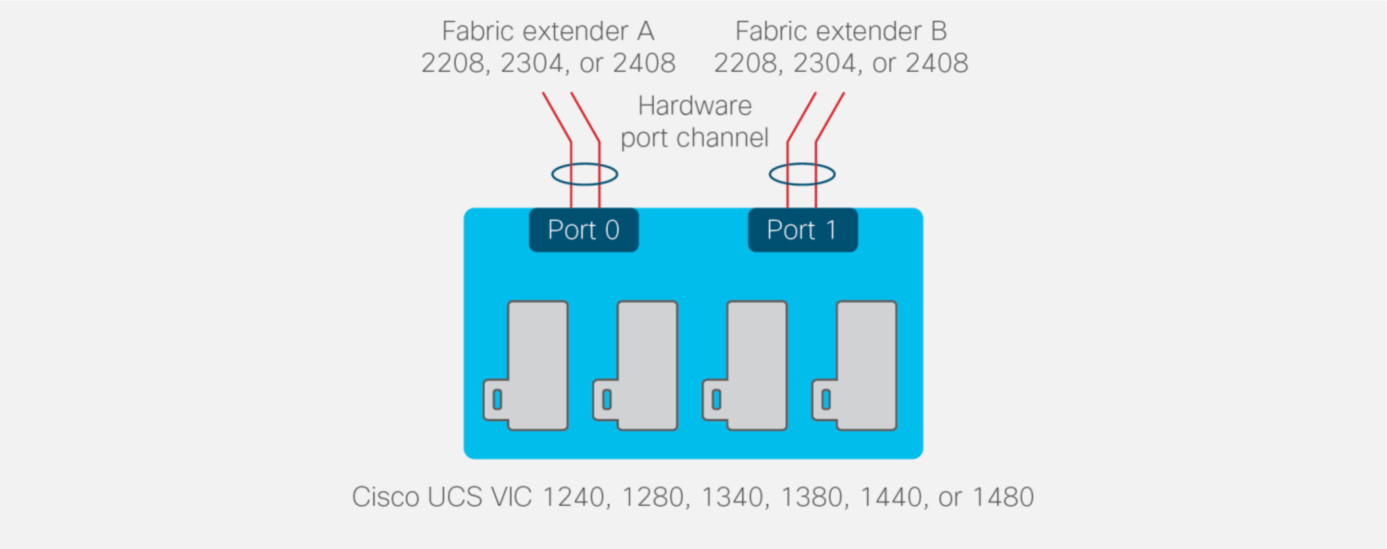


Figure 5. For higher-bandwidth Cisco UCS I/O modules, Cisco UCS VIC connections are combined through a hardware port channel

Table 1. vNIC bandwidth comparison based the combination of adapter and Cisco UCS fabric interconnect

Cisco UCS Blade adapter	Cisco UCS IOM	Physical connection	OS vNIC speed	Maximum bandwidth for a single flow[1]	Maximum aggregate bandwidth for vNIC
VIC 1240/1340/1440 or 1280/1380/1480	2204	10 Gbps	10 Gbps	10 Gbps	10 Gbps
VIC 1240/1340 Plus Port Expander	2204	2x10 Gbps in port-channel	20 Gbps	10 Gbps	20 Gbps[2]
VIC 1240/1340/1440 or 1280/1380/1480	2208	2x10 Gbps in port-channel	20 Gbps	10 Gbps	20 Gbps[2]
VIC 1240/1340 Plus Port Expander	2208	4x10 Gbps in port-channel	40 Gbps	10 Gbps	40 Gbps[2]
VIC 1240/1340/1440 or 1280/1380/1480	2304	2x10 Gbps in port-channel	20 Gbps	10 Gbps	20 Gbps[2]
VIC 1240 Plus Port Expander	2304	4x10 Gbps in port-channel	40 Gbps	10 Gbps	40 Gbps[2]
VIC 1340/1440 Plus Port Expander[3]	2304	40 Gbps	40 Gbps	40 Gbps	40 Gbps
VIC 1340/1440 or 1380/1480	2408	2x10 Gbps in port-channel	20 Gbps	10 Gbps	20 Gbps[2]
VIC 1340 Plus Port Expander	2408	4x10 Gbps in port-channel	40 Gbps	10 Gbps	40 Gbps[2]
VIC 1440 Plus Port Expander[4]	2408	40 Gbps	40 Gbps [4]	25 Gbps[4]	40 Gbps[4]

[1] Single TCP connection, i.e, single file transfer between two servers

[2] Multiple TCP connections where the flows are hashed across the physical connections, e.g., two or more file transfers either from the same two servers or from one to multiple servers

[3] There is an option to convert to a 4x 10-Gbps mode that operates similarly to the 1240 plus Port Expander mode. In 4x 10-Gbps mode, 1340 with the Port Expander operates as 4x 10-Gbps port-channels while the 1440 with the Port Expander operates as a 2x 10-Gbps port-channel.

[4] The vNIC single-flow maximum is 25 Gbps since the uplinks from the Cisco UCS 2408 Fabric Extender to the Cisco UCS 6400 Series Fabric Interconnects are 25 Gbps.

Adapter management

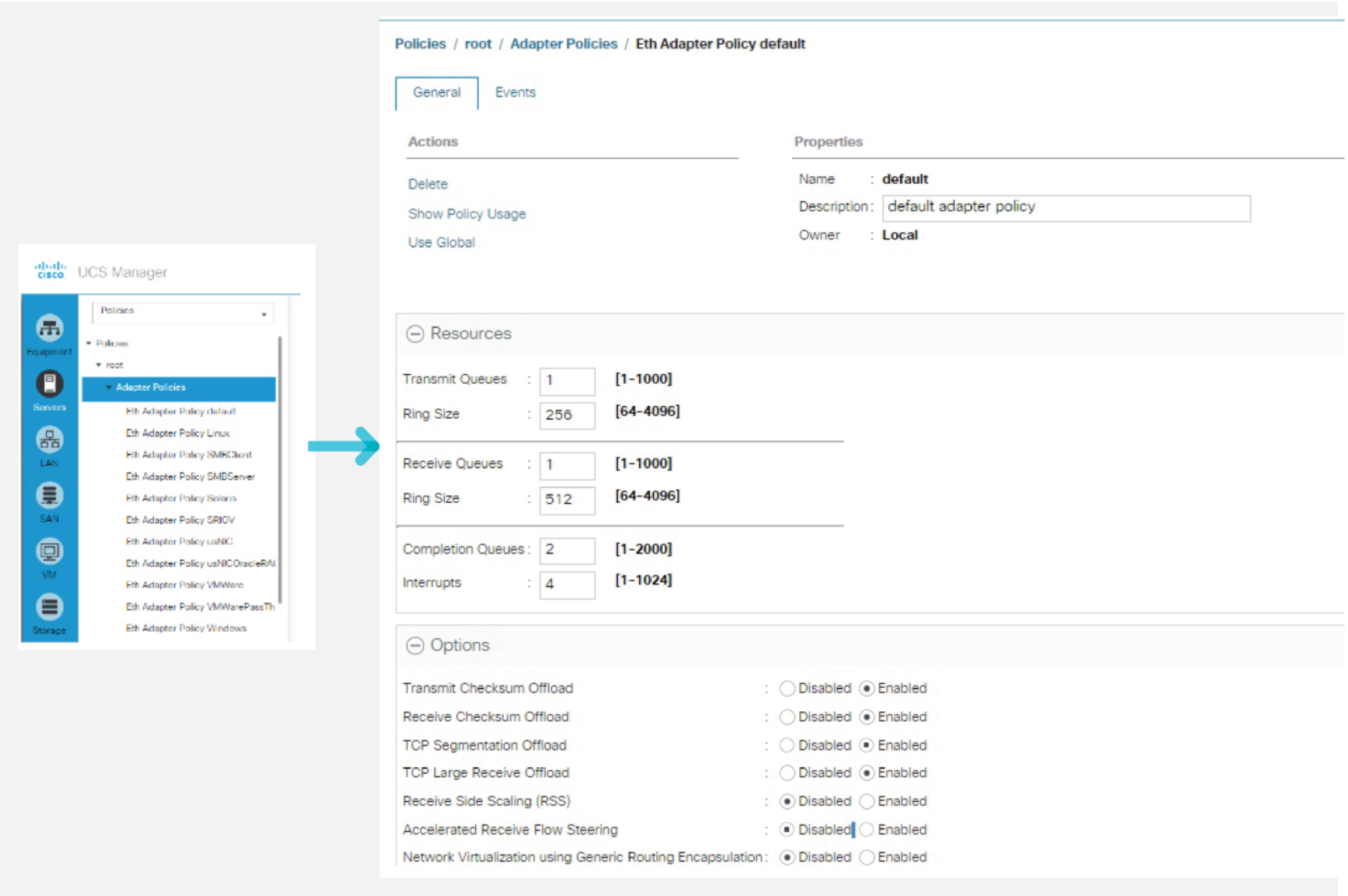
Cisco UCS servers provide a unique way to manage the adapter parameters through a programmable interface. The vNIC parameters for the adapter—offloads, queues, interrupts, etc.—are not configured through the OS. Instead, vNIC parameters are adjusted through a Cisco management tool set, using the Cisco UCS management GUI, XML API, or command-line interface (CLI).

Unified management with Cisco UCS Manager (rack and blade servers)

When Cisco UCS servers are connected to a fabric interconnect, Cisco UCS Manager becomes the single place for defining server policies and provisioning all software and hardware components across multiple blade and rack servers. Cisco UCS Manager uses service profiles to provision servers according to policy. The adapter policy is one of the components of the service profile.

The adapter policy provides significant advantages, because it can be defined once and then referenced in multiple different policies and profiles. This capability provides ease of deployment and operation and helps ensure consistency across servers and, potentially, between rack and blade platforms. The adapter policy allows the administrator to declare the capabilities of the vNIC, such as the number of rings, ring sizes, and offload enablement and disablement.

A default adapter policy is always defined and used by the vNIC. However, the administrator can change the default policy to one of the recommended OS default settings or to a newly defined adapter policy, depending on workload requirements. Figure 6 shows the adapter policies that can be selected in the Cisco UCS Manager GUI. Table 2 provides default recommended eNIC settings. Additional guidance about the use of these parameters is provided in the sections that follow.


UCS Manager
Equipment
Servers
LAN
SAN
VM
Storage
Policies
Policies
root
Adapter Policies
Eth Adapter Policy default
Eth Adapter Policy Linux
Eth Adapter Policy SMBClient
Eth Adapter Policy SMBServer
Eth Adapter Policy Solaris
Eth Adapter Policy SRIOV
Eth Adapter Policy usNIC
Eth Adapter Policy usNICOracleRA(
Eth Adapter Policy VMWare
Eth Adapter Policy VMWarePassTh
Eth Adapter Policy Windows
General
Events
Actions
Delete
Show Policy Usage
Use Global

Policies / root / Adapter Policies / **Eth Adapter Policy default**

Properties

Name : **default**

Description : default adapter policy

Owner : **Local**

Resources

Transmit Queues :	1	[1-1000]
Ring Size :	256	[64-4096]
Receive Queues :	1	[1-1000]
Ring Size :	512	[64-4096]
Completion Queues :	2	[1-2000]
Interrupts :	4	[1-1024]

Options

Transmit Checksum Offload :	○ Disabled	● Enabled
Receive Checksum Offload :	○ Disabled	● Enabled
TCP Segmentation Offload :	○ Disabled	● Enabled
TCP Large Receive Offload :	○ Disabled	● Enabled
Receive Side Scaling (RSS) :	● Disabled	○ Enabled
Accelerated Receive Flow Steering :	● Disabled	○ Enabled
Network Virtualization using Generic Routing Encapsulation :	● Disabled	○ Enabled

Figure 6. Cisco UCS Manager lets administrators choose and adjust the adapter policy for the Cisco UCS VIC

Table 2. Recommended default OS eNIC settings

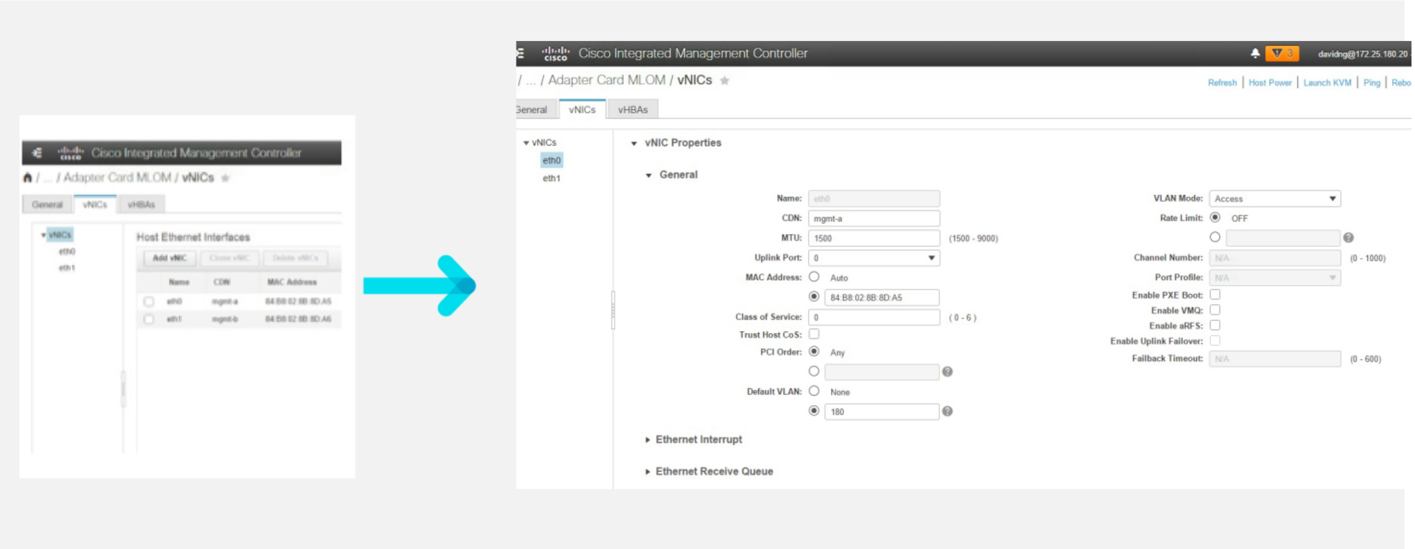
Parameter	Default	Microsoft Windows	Linux	VMware ESXi
Transmit (TX) settings (number of queues and ring size)	Queues: 1 Ring size: 256	Queues: 1 Ring size: 256	Queues: 1 Ring size: 256	Queues: 1 Ring size: 256
Receive (RX) settings (number of queues and ring size)	Queues: 1 Ring size: 512	Queues: 4 Ring size: 512	Queues: 1 Ring size: 512	Queues: 1 Ring size: 512
Number of completion queues (CQs) and interrupts	CQs: 2 Interrupts: 4	CQs: 5 Interrupts: 8 or 512	CQs: 2 Interrupts: 4	CQs: 2 Interrupts: 4
Adapter options	TX and RX checksum, TCP segmentation offload (TSO), and large receive offload (LRO)	TX and RX checksum, TSO, LRO, and receive-side scaling (RSS)	TX and RX checksum, TSO, and LRO	TX and RX checksum, TSO, and LRO

Note: For Microsoft Windows operating systems, the default adapter policy in Cisco UCS Manager for 1400 series VICs is "Win-HPN," which uses an interrupt of 512.

Standalone management (rack servers)

When a Cisco UCS C-Series Rack Server is not being managed by Cisco UCS Manager, the Cisco IMC provides embedded server management and monitoring. Similar to Cisco UCS Manager, Cisco IMC provides programmatic control for several of the server components, such as BIOS settings, boot order, storage management, and vNIC settings. In addition, the parameters and settings can be configured through the Cisco IMC GUI, the CLI, or the XML API.

When a Cisco UCS C-Series Rack Server is in standalone mode, two vNICs are created by default (Figure 7), and they cannot be deleted. Up to 16 additional vNICs can be created as needed, with each vNIC having its own unique settings.



	Name	CDN	MAC Address
□	eth0	mgmt-a	84:B8:02:8B:8D:A5
□	eth1	mgmt-b	84:B8:02:8B:8D:A6

Name: eth0
CDN: mgmt-a
MTU: 1500 (1500 - 9000)
Uplink Port: 0
MAC Address: ○ Auto
● 84:B8:02:8B:8D:A5
Class of Service: 0 (0 - 6)
Trust Host CoS: □
PCI Order: ● Any
○
Default VLAN: ○ None
● 180

VLAN Mode: Access
Rate Limit: ● OFF
○
Channel Number: N/A (0 - 1000)
Port Profile: N/A
Enable PXE Boot: □
Enable VMQ: □
Enable aRFS: □
Enable Uplink Failover: □
Failback Timeout: N/A (0 - 600)

Figure 7. vNIC properties displayed in the Cisco IMC GUI

The default settings for the two initially created vNICs and for any newly created vNICs are shown in Table 3. These initial settings can be altered to better suit individual application requirements.

Table 3. Default vNIC settings

Parameter	Default
TX settings **(number of queues and ring size)**	Queues: 1 Ring size: 256
RX settings **(number of queues and ring size)**	Queues: 4 Ring size: 512
Number of CQs and interrupts	CQs: 5 Interrupts: 8
Adapter options	TX and RX checksum, TSO, LRO, and RSS

If multiple vNICs are using the same port interface, the bandwidth is shared equitably between the vNICs. Each vNIC can use the available bandwidth. If all the vNICs are vying for bandwidth, then bandwidth is equally divided among the vNICs. For example, if Port 0 is servicing three vNICs that are all part of the same quality-of-service (QoS) policy, each vNIC will get an equal share (33 percent) of the available bandwidth.

In some instances, you may want to cap the upper bandwidth limit for a particular vNIC. For example, live migration is bandwidth intensive and tends to consume as much bandwidth as is available. Although live migration is important, you may want to rate-limit vNIC traffic in hardware to cap bandwidth consumption, to help ensure that other vNIC workloads get the bandwidth they need.

In Figures 8 and 9, a rate limit is created through a QoS policy for the vNIC dedicated to live migration (VMware vMotion traffic), capping bandwidth at 2 Gbps. As a result, the vNIC can never consume more than 2 Gbps. Assuming a 10-Gbps interface, the other two vNICs (serving data and management traffic) will share the remaining 8 Gbps. If the vMotion vNIC is not active, the data and management vNICS will be able to use the full 10 Gbps of bandwidth.

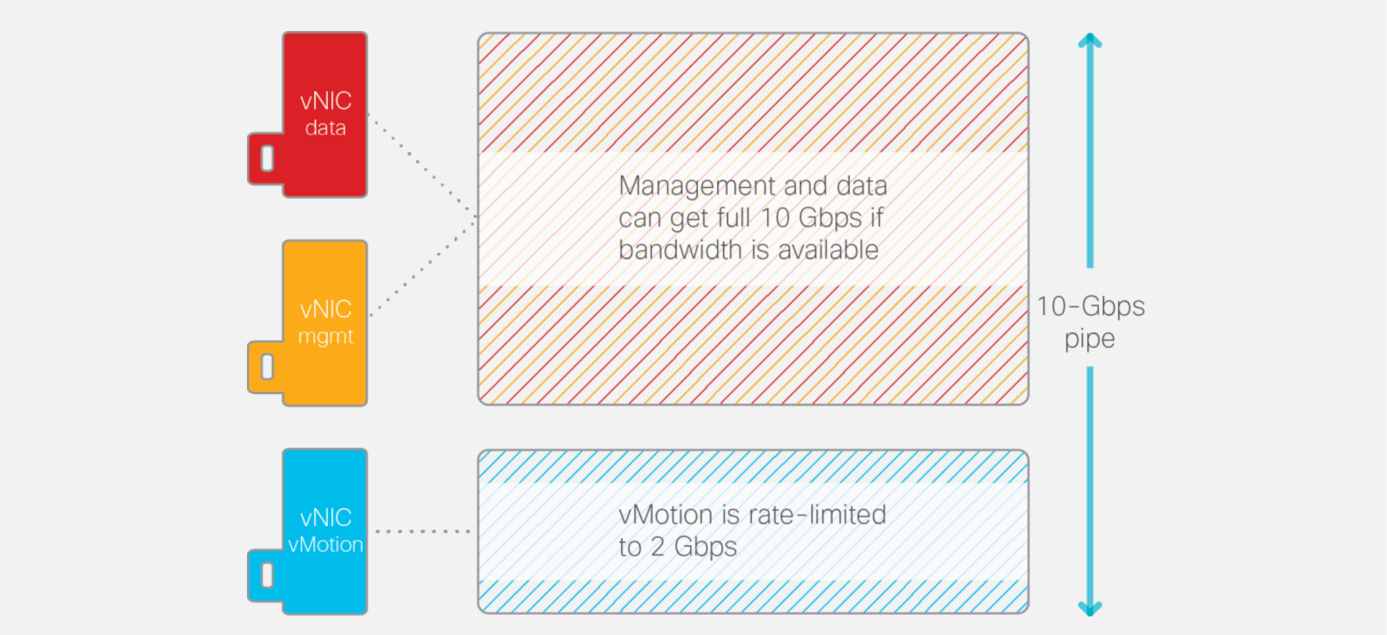
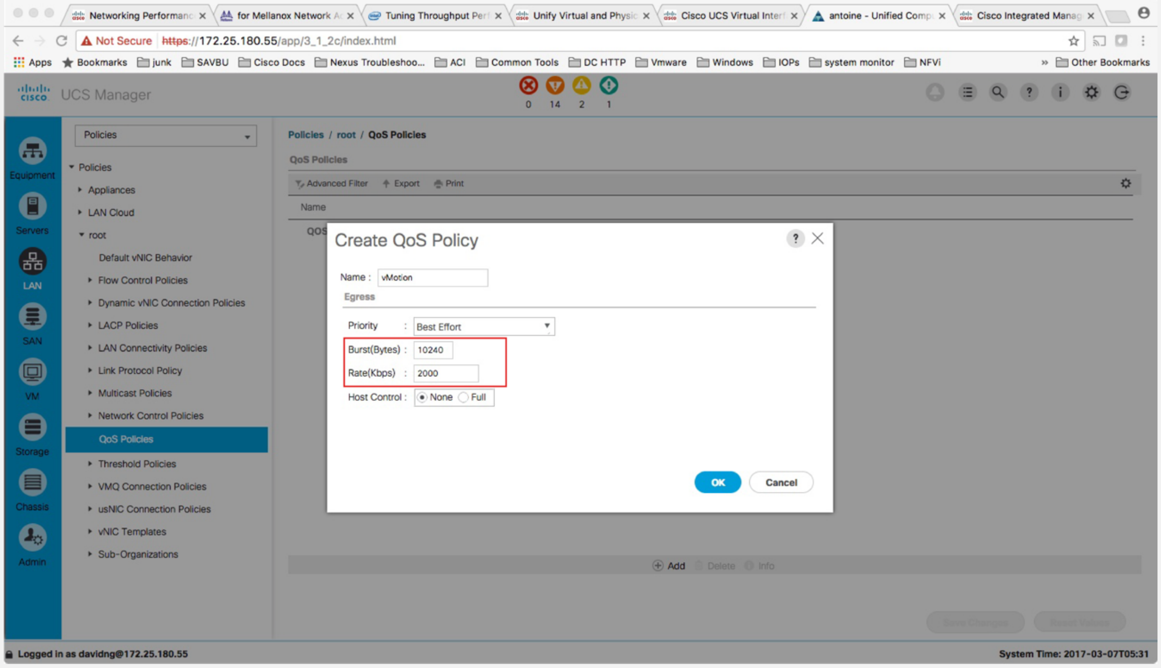

vNIC
data
vNIC
mgmt
vNIC
vMotion
Management and data can get full 10 Gbps if bandwidth is available
vMotion is rate-limited to 2 Gbps
10-Gbps pipe

Figure 8. A vNIC can be rated-limited to prevent it from consuming all the available bandwidth



Create QoS Policy

Name : vMotion

Egress

Priority : Best Effort

Burst(Bytes) : 10240

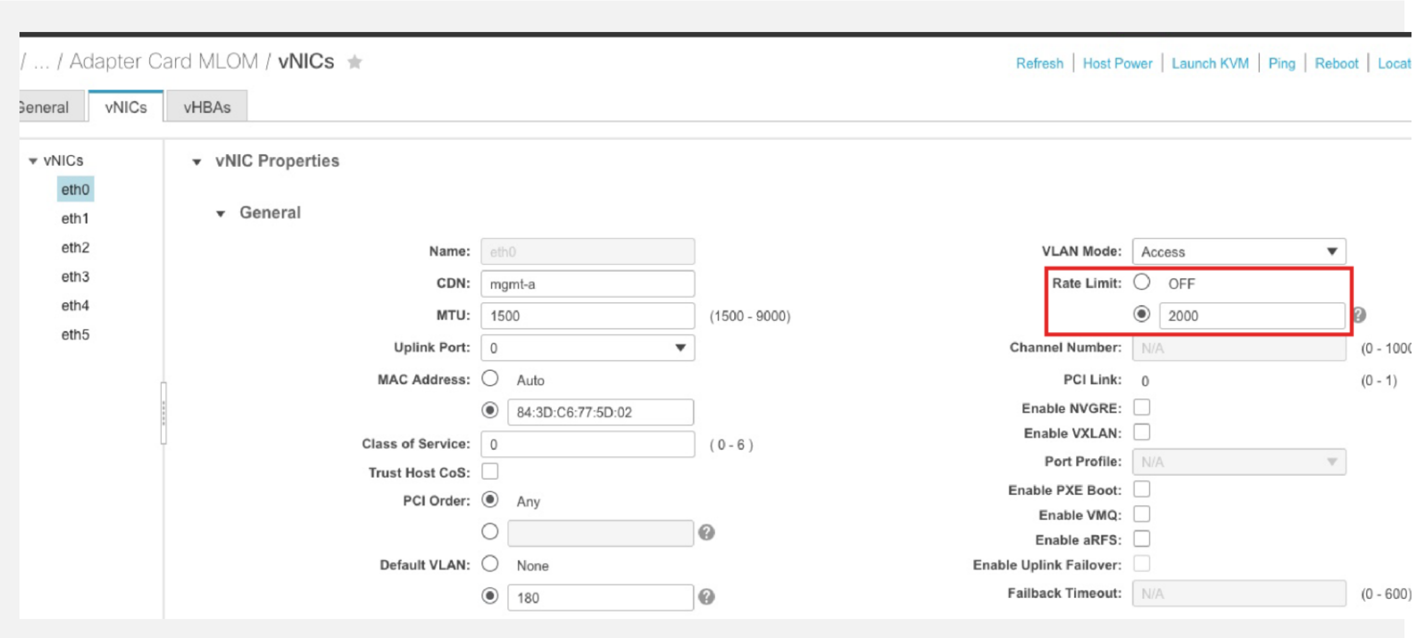
Rate(Kbps) : 2000

Host Control : ● None ○ Full

OK Cancel

Figure 9. A QoS policy creates a rate limit for a specific vNIC

Figure 10 shows a QoS policy being instantiated for a vNIC through the Cisco IMC GUI.


/ ... / Adapter Card MLOM / vNICs ★
Refresh | Host Power | Launch KVM | Ping | Reboot | Locat
General
vNICs
vHBAs
▾ vNICs
eth0
eth1
eth2
eth3
eth4
eth5
▾ vNIC Properties
▾ General

Name: eth0
CDN: mgmt-a
MTU: 1500 (1500 - 9000)
Uplink Port: 0
MAC Address: ○ Auto
● 84:3D:C6:77:5D:02
Class of Service: 0 (0 - 6)
Trust Host CoS: □
PCI Order: ● Any
○
Default VLAN: ○ None
● 180

VLAN Mode: Access
Rate Limit: ○ OFF
● 2000
Channel Number: N/A (0 - 100(
PCI Link: 0 (0 - 1)
Enable NVGRE: □
Enable VXLAN: □
Port Profile: N/A
Enable PXE Boot: □
Enable VMQ: □
Enable aRFS: □
Enable Uplink Failover: □
Failback Timeout: N/A (0 - 600)

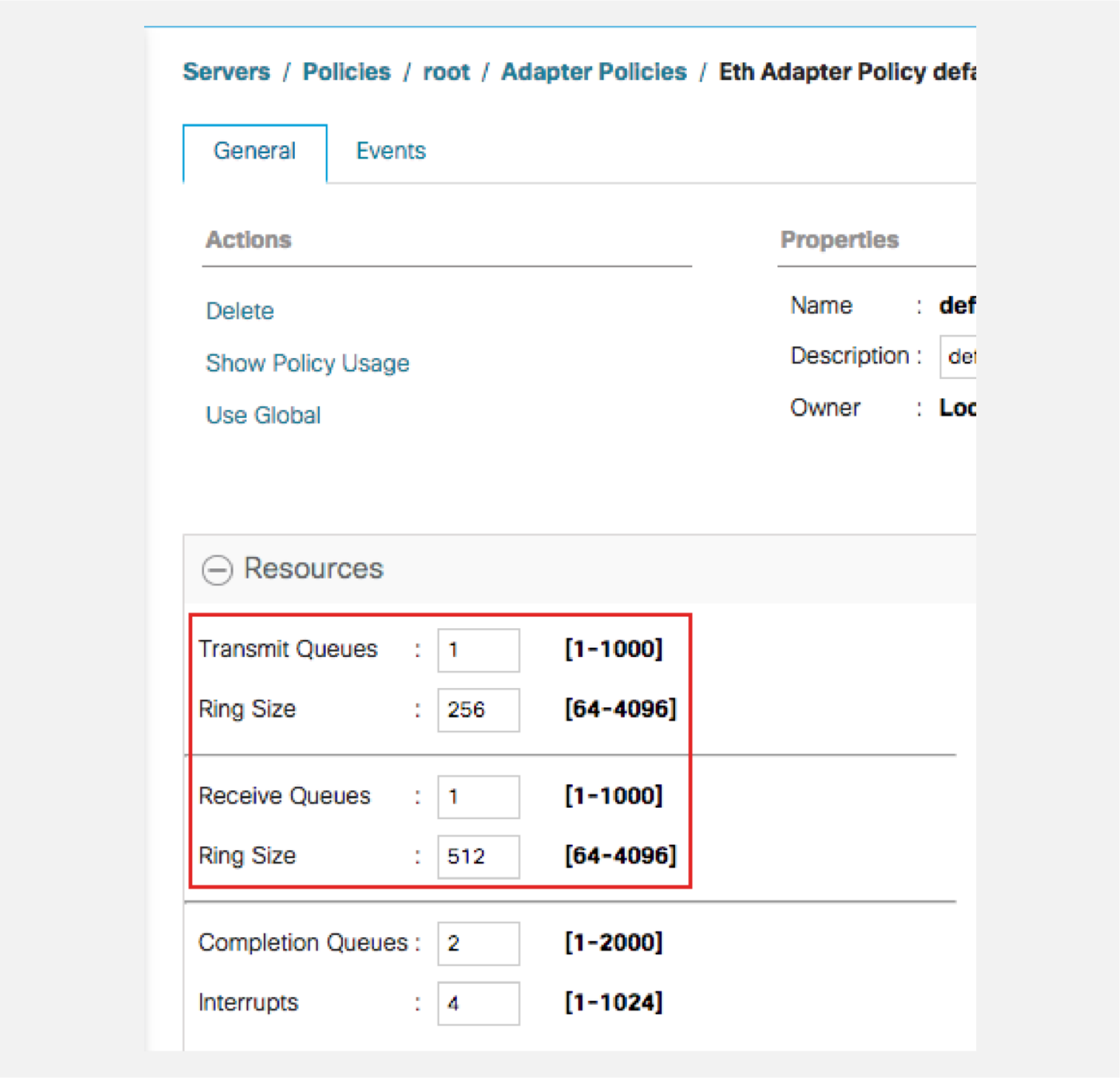
Figure 10. Rate limiting by QoS policy through the Cisco IMC

Tunable eNIC parameters

The sections that follow describe a number of eNIC parameters that you can tune to optimize performance for the vNIC.

Transmit and receive queues

The adapter policy allows you to define the number of queues for transmit and receive traffic on a per-vNIC basis. Increasing the number of queues enables simultaneous transmit and receive packets on multiple ring queues from multiple CPU cores. The number of transmit and receive queues can be defined independent of each other. For example, a vNIC can be configured with one transmit queue and four receive queues. Figures 11 and 12 show transmit and receive queue count and ring size configurations using Cisco UCS Manager and Cisco IMC respectively.



Servers / Policies / root / Adapter Policies / Eth Adapter Policy defa
General
Events
Actions
Delete
Show Policy Usage
Use Global
Properties

Name : def

Description : def

Owner : Loc

Resources

Transmit Queues : 1 [1-1000]

Ring Size : 256 [64-4096]

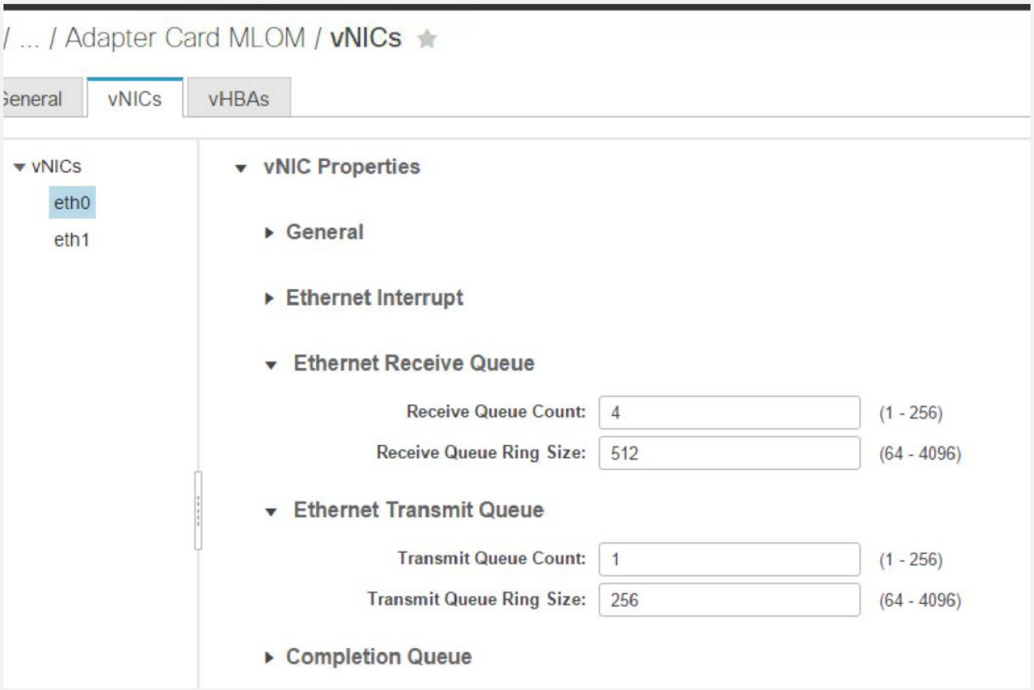
Receive Queues : 1 [1-1000]

Ring Size : 512 [64-4096]

Completion Queues : 2 [1-2000]

Interrupts : 4 [1-1024]

Figure 11. Configuring adapter policies for transmit and receive queues using Cisco UCS Manager



Ethernet Receive Queue

Receive Queue Count:	4	(1 - 256)
Receive Queue Ring Size:	512	(64 - 4096)

Ethernet Transmit Queue

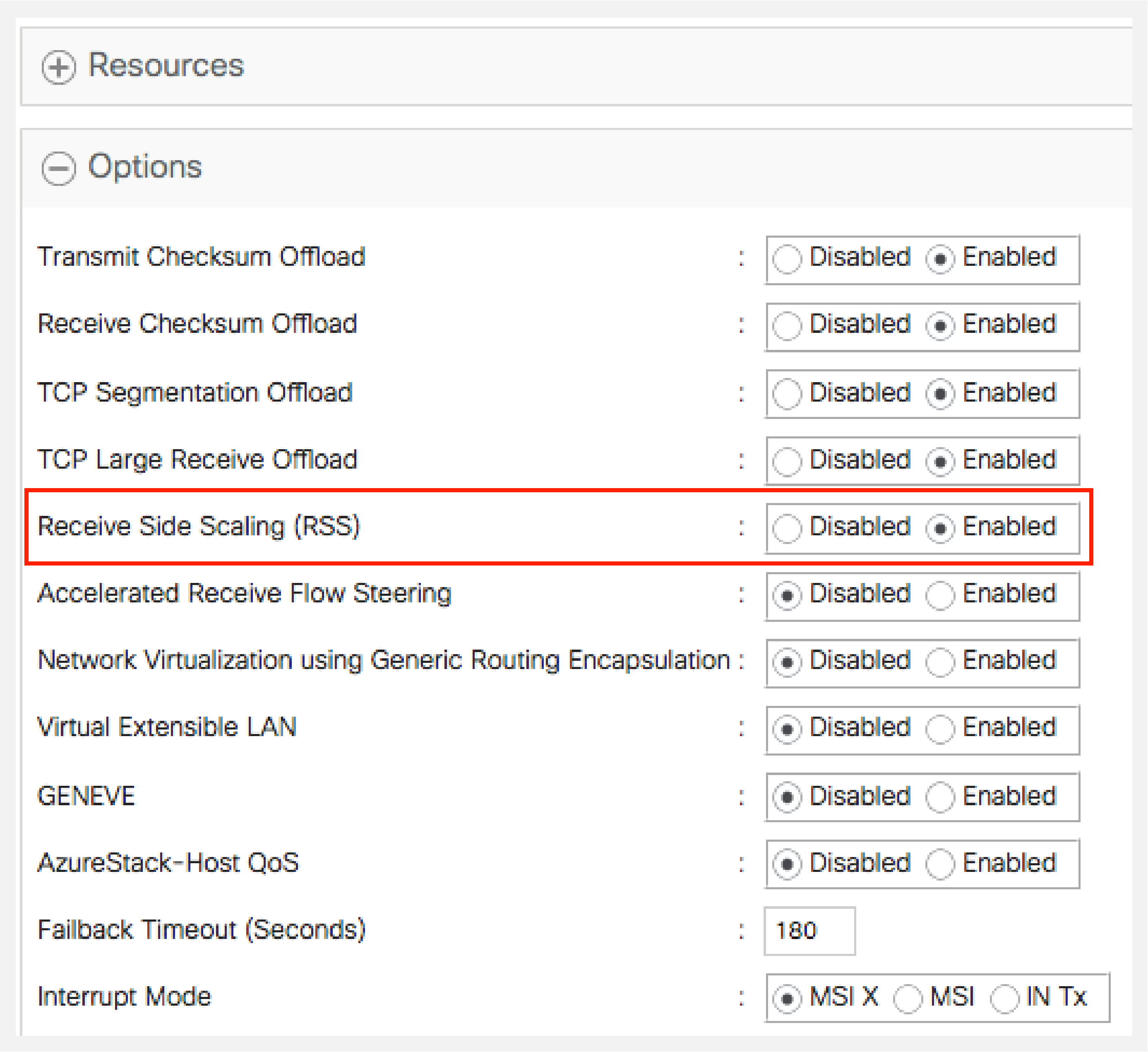
Transmit Queue Count:	1	(1 - 256)
Transmit Queue Ring Size:	256	(64 - 4096)

Note: If you configure more than one receive queue, you need to enable RSS for the adapter policy.

Figure 12. Configuring transmit and receive queues with Cisco IMC

Receive-side scaling

The Cisco UCS VIC supports RSS, and it can be enabled on a per-vNIC basis. RSS allows the incoming traffic to be spread across multiple CPU cores. The CPU cores can then pull the packets in parallel, helping improve the overall bandwidth to the server. Figures 13 and 14 illustrate how to enable RSS using Cisco UCS Manager and Cisco IMC respectively.



Transmit Checksum Offload : ○ Disabled ● Enabled

Receive Checksum Offload : ○ Disabled ● Enabled

TCP Segmentation Offload : ○ Disabled ● Enabled

TCP Large Receive Offload : ○ Disabled ● Enabled

Receive Side Scaling (RSS) : ○ Disabled ● Enabled

Accelerated Receive Flow Steering : ● Disabled ○ Enabled

Network Virtualization using Generic Routing Encapsulation : ● Disabled ○ Enabled

Virtual Extensible LAN : ● Disabled ○ Enabled

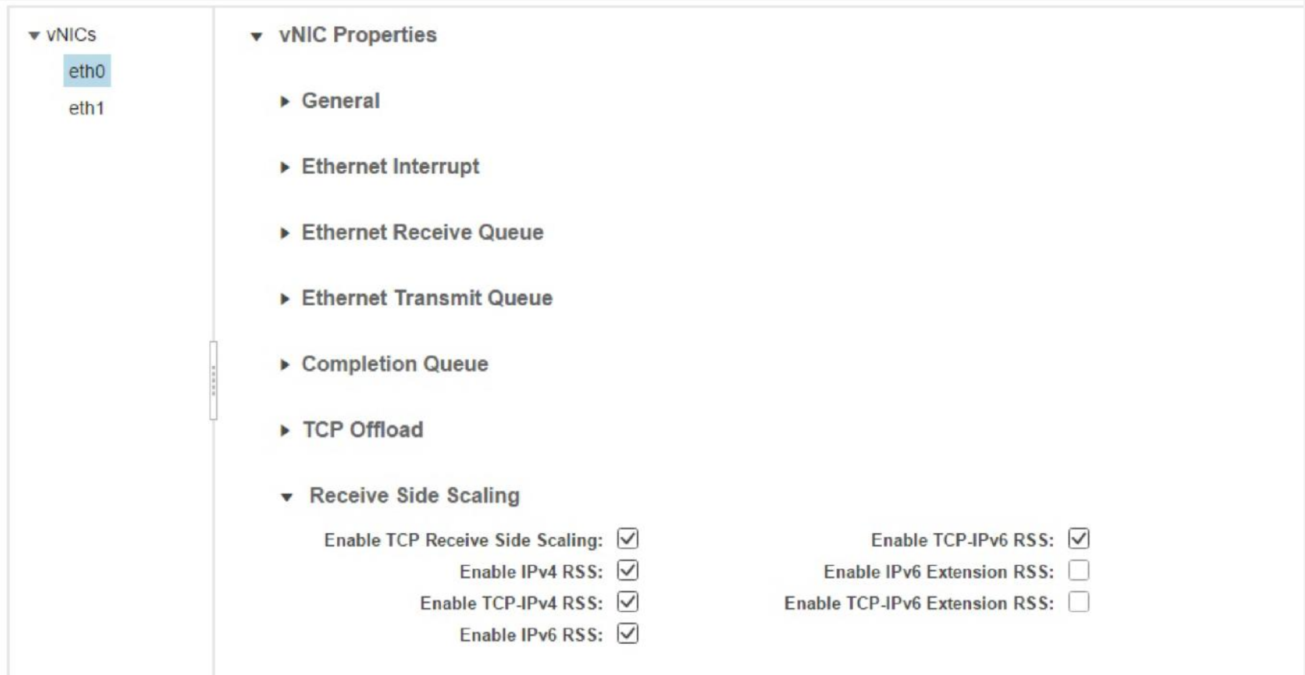
GENEVE : ● Disabled ○ Enabled

AzureStack-Host QoS : ● Disabled ○ Enabled

Failback Timeout (Seconds) : 180

Interrupt Mode : ● MSI X ○ MSI ○ IN Tx

Figure 13. Enabling RSS with Cisco UCS Manager



Enable TCP Receive Side Scaling: ☑
Enable IPv4 RSS: ☑
Enable TCP-IPv4 RSS: ☑
Enable IPv6 RSS: ☑
Enable TCP-IPv6 RSS: ☑
Enable IPv6 Extension RSS: □
Enable TCP-IPv6 Extension RSS: □

The implementation of RSS differs slightly between Cisco UCS Manager and Cisco IMC. When RSS is enabled in Cisco UCS Manager, both IPv4 and IPv6 protocols are enabled. When RSS is enabled using Cisco IMC, protocol selection is allowed.

Figure 14. Enabling RSS with Cisco IMC

RSS needs to be enabled when more than a single receive queue is configured, to spread the load across multiple CPU cores. The traffic distribution across the receive queue is based on the IP address (IPv4 or IPv6) and TCP port numbers. Therefore, a single flow is directed to the same receive queue to help ensure in-order delivery. The maximum number of receive queues supported is OS specific. Table 4 lists the maximum number of receive queues supported on different OS platforms. Note that in Table 4, depending on the Linux VIC enic driver version, the maximum number of RX queues are different: 8 RX queues per vNIC for enic driver version up to 3.2 and up to the number of CPU cores in the server for enic driver version 3.2 and above.

Table 4. RSS and maximum supported receive queues

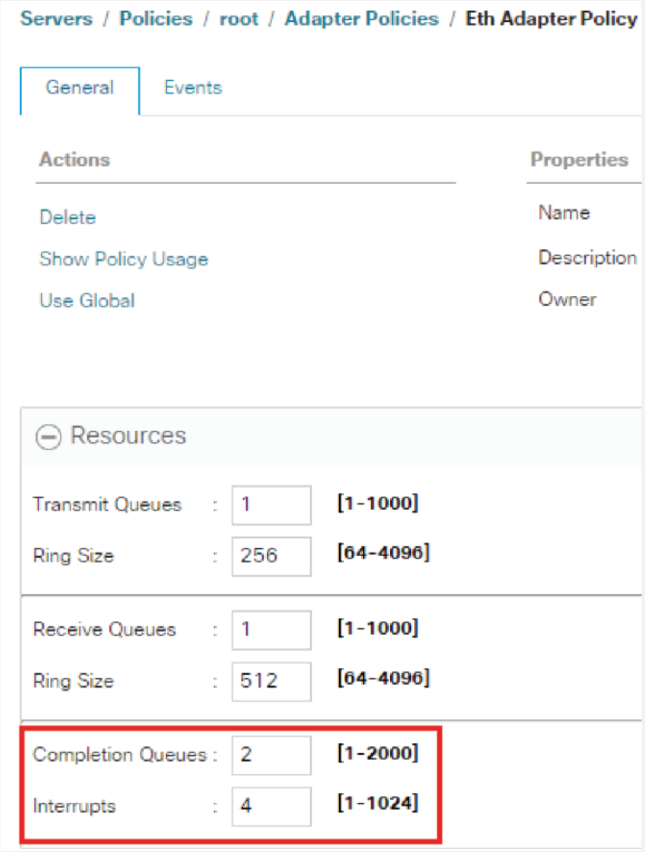
Parameteer	Microsoft Windows	Linux	VMware ESXi
RX queue	8	8 or number of CPU cores	16
RSS	Enabled	Enabled	Enabled

Transmit and receive queue sizes

The Cisco UCS VIC allows manual configuration of transmit and receive ring queue buffer descriptors on a per-vNIC basis. The default buffers are set small for lower packet latency. However, in some cases increasing the buffer size is warranted. If the network load is heavy and dropped packets are observed on the adapter interface, increasing the queue buffer can help reduce the number of dropped packets and improve overall performance.

Completion queues and interrupts

Completion queues are used to notify software of packet transmit and packet receive events. The completion queue can be dedicated to an individual transmit or receive queue. Alternatively, multiple transmit and receive queues can share a single completion queue. The vNIC uses interrupts to notify the CPU that a packet is ready to be picked up from the kernel buffer. Figures 15 and 16 show completion queue and interrupt configuration using Cisco UCS Manager and Cisco IMC respectively.

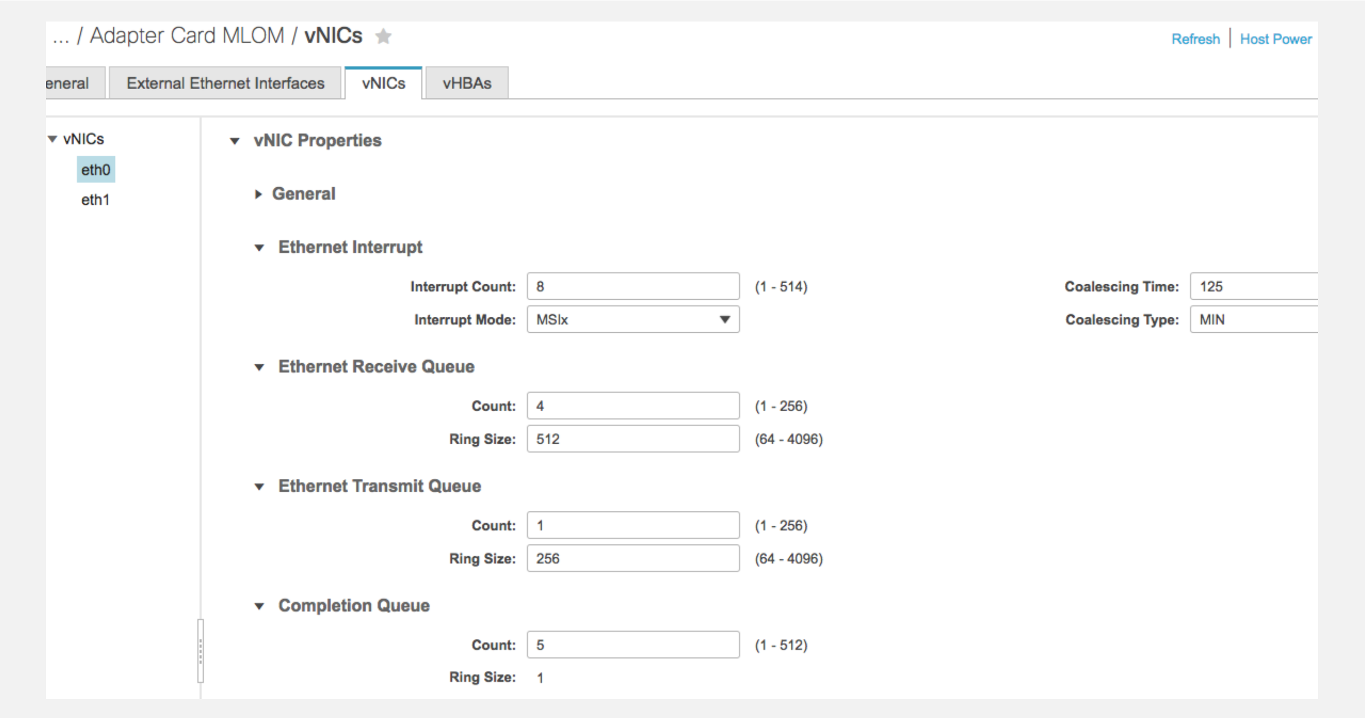


Transmit Queues : 1 [1-1000]
Ring Size : 256 [64-4096]

Receive Queues : 1 [1-1000]
Ring Size : 512 [64-4096]

Completion Queues : 2 [1-2000]
Interrupts : 4 [1-1024]

Figure 15. Completion queue and interrupt configuration for Cisco UCS Manager



Ethernet Interrupt

Interrupt Count: 8 (1 - 514)

Interrupt Mode: MSIx

Coalescing Time: 125

Coalescing Type: MIN

Ethernet Receive Queue

Count: 4 (1 - 256)

Ring Size: 512 (64 - 4096)

Ethernet Transmit Queue

Count: 1 (1 - 256)

Ring Size: 256 (64 - 4096)

Completion Queue

Count: 5 (1 - 512)

Ring Size: 1

Figure 16. Completion queue and interrupt configuration for Cisco IMC

The number of completion queues and number of interrupts have a direct relationship with the number of transmit and receive queues. The number of completion queues allocated should be the sum of the transmit queues and receive queues (CQ = TXQ + RXQ). The interrupt calculation, however, varies by operating system.

- For Linux, the number of interrupts is set to “CQ + 2” for the Cisco UCS Linux enic driver version up to 3.2. For enic driver 3.2 and up the interrupt would be “(maximum of TX or RX queue) + 2.”
- For Microsoft Windows, the number of interrupts for 1200 and 1300 series VICs is “CQ + 2 rounded up to the nearest power of 2” and for the 1400 series VICs the interrupt would be at least “2x server_cpu-cores + 4” or up to a maximum of 512.
- For ESXi, the interrupt would be set as “CQ + 2”.

Table 5. Recommended default OS eNIC settings

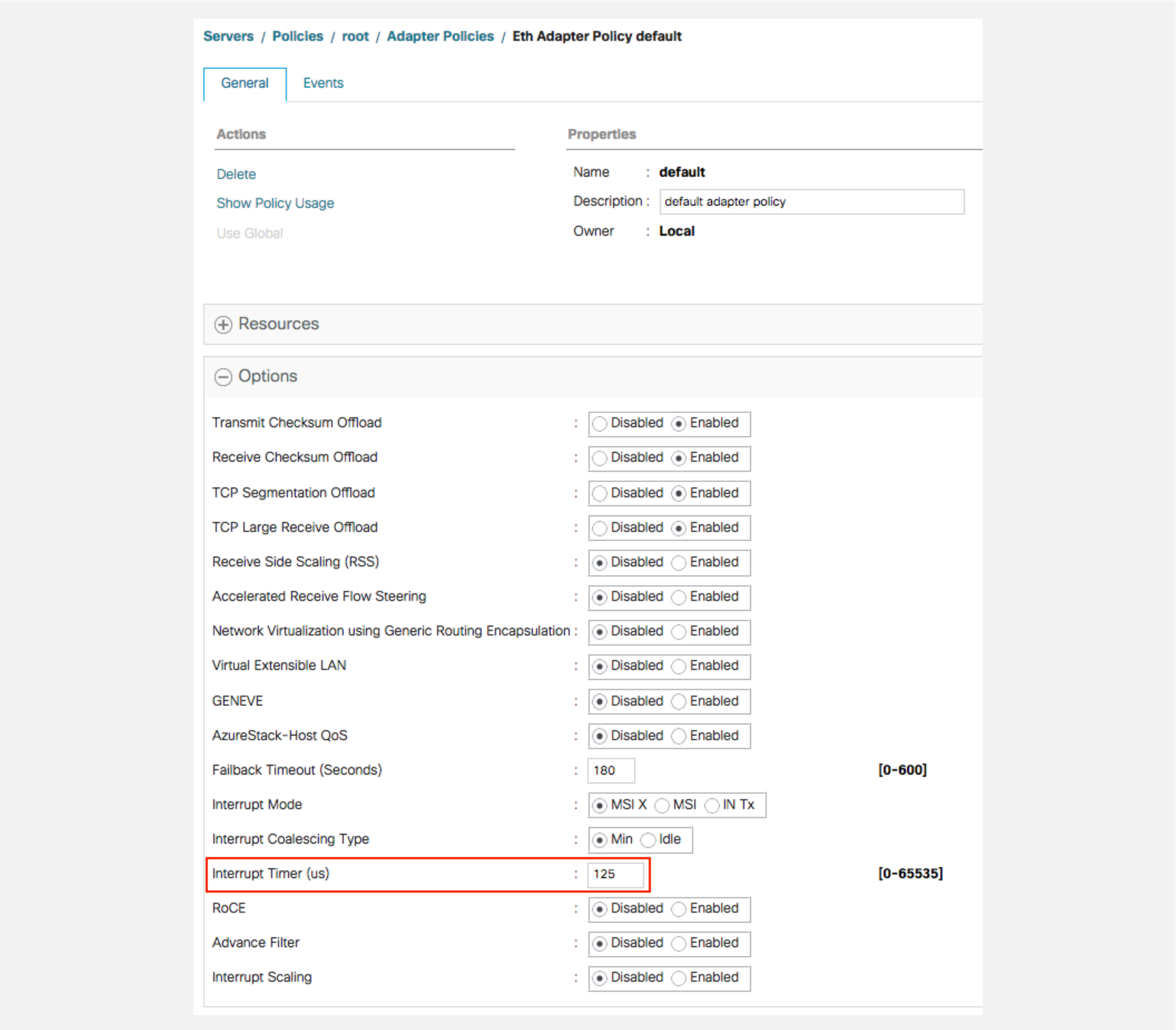
Parameter	1 TX queue and 4 RX queues		4 TX queues and 8 RX queues	
	Linux	Microsoft Windows	Linux	Microsoft Windows
CQ	5	5	12	12
Interrupt	6 or 7[1]	8 or 512[2]	10 or 14[1]	16 or 512[2]

[1] Interrupt value of 7 or 14 (CQ + 2) for linux enic driver below 3.2 and 6 or 10 (Maximum of TX/RX + 2) for enic drivers above 3.2

[2] For the Cisco UCS VIC 1400 Series, the recommended interrupt value should be at least “2x CPU core + 4” or up to a maximum 512

Interrupt coalescing timer

As mentioned earlier, hardware interrupts are raised to inform the CPU to either pull the packet from the receive queue or transmit the packet to the wire. The interrupt rate can be controlled with different types of wait periods. This value changes both the transmit timer and the receive timer. In most cases, the timer does not need to be adjusted. For latency-sensitive workloads or intensive transactional applications, however, lowering the rate can increase performance. However, a lower rate will also increase CPU utilization and can potentially interfere with other processes. Figures 17 and 18 show interrupt coalescing timer configuration using Cisco UCS Manager and Cisco IMC respectively.



Servers / Policies / root / Adapter Policies / Eth Adapter Policy default

Properties

Name : **default**

Description : default adapter policy

Owner : **Local**

Options

Transmit Checksum Offload : ○ Disabled ● Enabled

Receive Checksum Offload : ○ Disabled ● Enabled

TCP Segmentation Offload : ○ Disabled ● Enabled

TCP Large Receive Offload : ○ Disabled ● Enabled

Receive Side Scaling (RSS) : ● Disabled ○ Enabled

Accelerated Receive Flow Steering : ● Disabled ○ Enabled

Network Virtualization using Generic Routing Encapsulation : ● Disabled ○ Enabled

Virtual Extensible LAN : ● Disabled ○ Enabled

GENEVE : ● Disabled ○ Enabled

AzureStack-Host QoS : ● Disabled ○ Enabled

Failback Timeout (Seconds) : 180 **[0-600]**

Interrupt Mode : ● MSI X ○ MSI ○ IN Tx

Interrupt Coalescing Type : ● Min ○ Idle

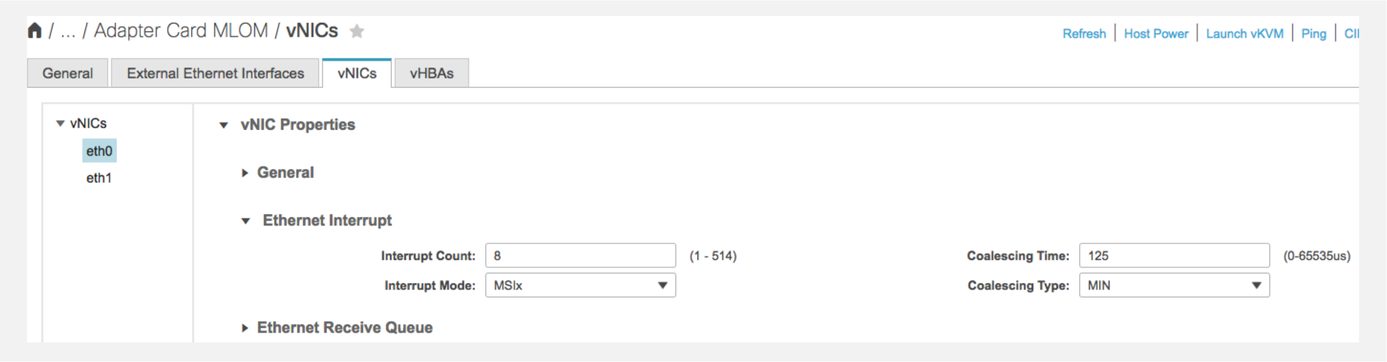
Interrupt Timer (us) : 125 **[0-65535]**

RoCE : ● Disabled ○ Enabled

Advance Filter : ● Disabled ○ Enabled

Interrupt Scaling : ● Disabled ○ Enabled

Figure 17. Setting the interrupt timer with Cisco UCS Manager


/ ... / Adapter Card MLOM / vNICs
Refresh | Host Power | Launch vKVM | Ping | CI
General
External Ethernet Interfaces
vNICs
vHBAs
vNICs
eth0
eth1

vNIC Properties

General

Ethernet Interrupt

Interrupt Count: 8 (1 - 514)

Interrupt Mode: MSIx

Coalescing Time: 125 (0-65535us)

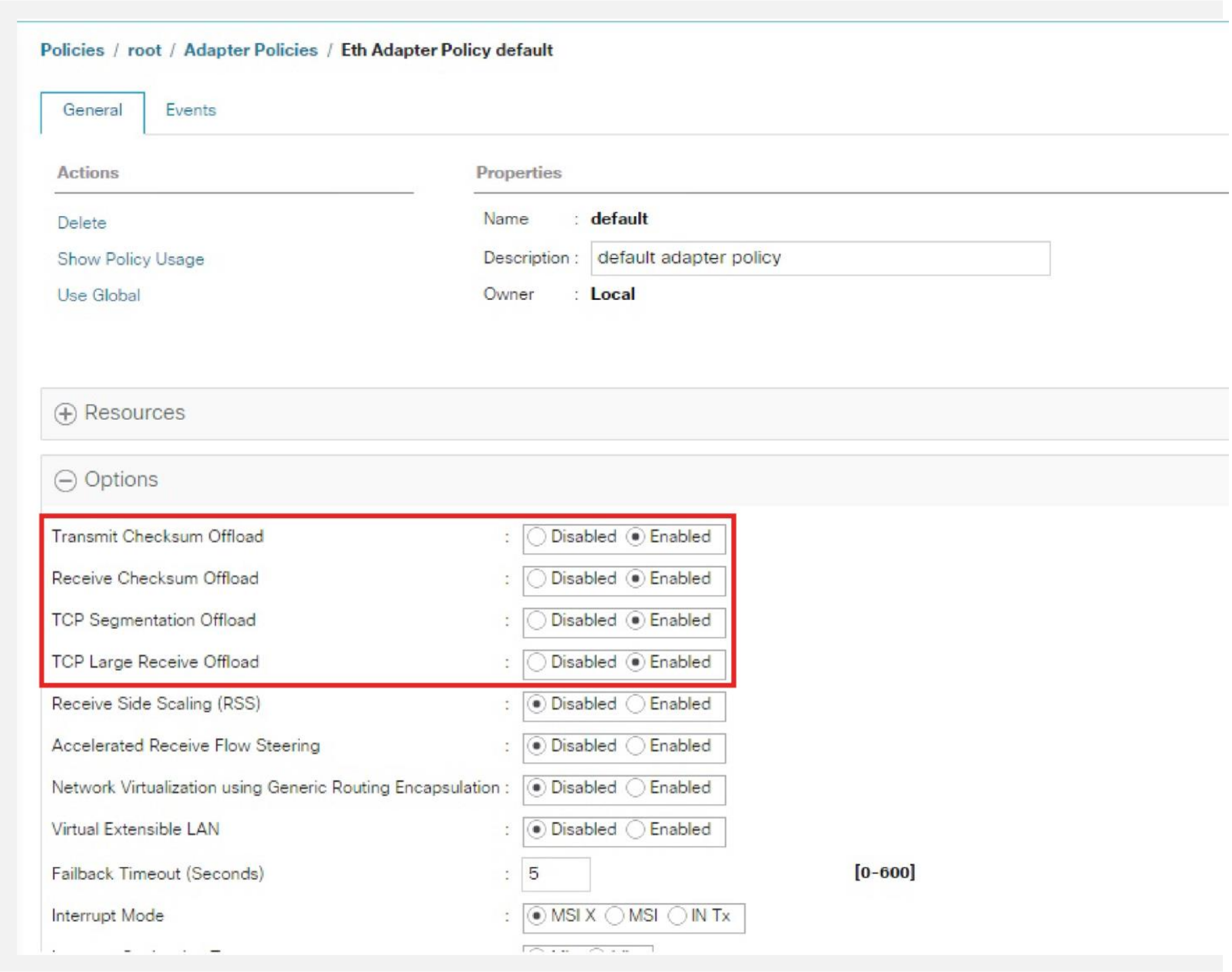
Coalescing Type: MIN

Ethernet Receive Queue

Figure 18. Setting the interrupt timer with Cisco IMC

Offloading

Offloading is a beneficial feature that should be enabled in almost all cases. Offloading allows the OS to take advantage of the adapter hardware to perform certain functions (Table 6), reducing the CPU processing requirements for these specific packet-level functions. By default, most of the offloads supported by the Cisco UCS VIC are enabled for both IPv4 and IPv6 packets, including transmit and receive checksum offload, TCP segmentation offload, and TCP large receive offload. Figures 19 and 20 show offload configuration using Cisco UCS Manager and Cisco IMC respectively.



Policies / root / Adapter Policies / Eth Adapter Policy default

Name : **default**

Description : default adapter policy

Owner : **Local**

Transmit Checksum Offload : ○ Disabled ● Enabled

Receive Checksum Offload : ○ Disabled ● Enabled

TCP Segmentation Offload : ○ Disabled ● Enabled

TCP Large Receive Offload : ○ Disabled ● Enabled

Receive Side Scaling (RSS) : ● Disabled ○ Enabled

Accelerated Receive Flow Steering : ● Disabled ○ Enabled

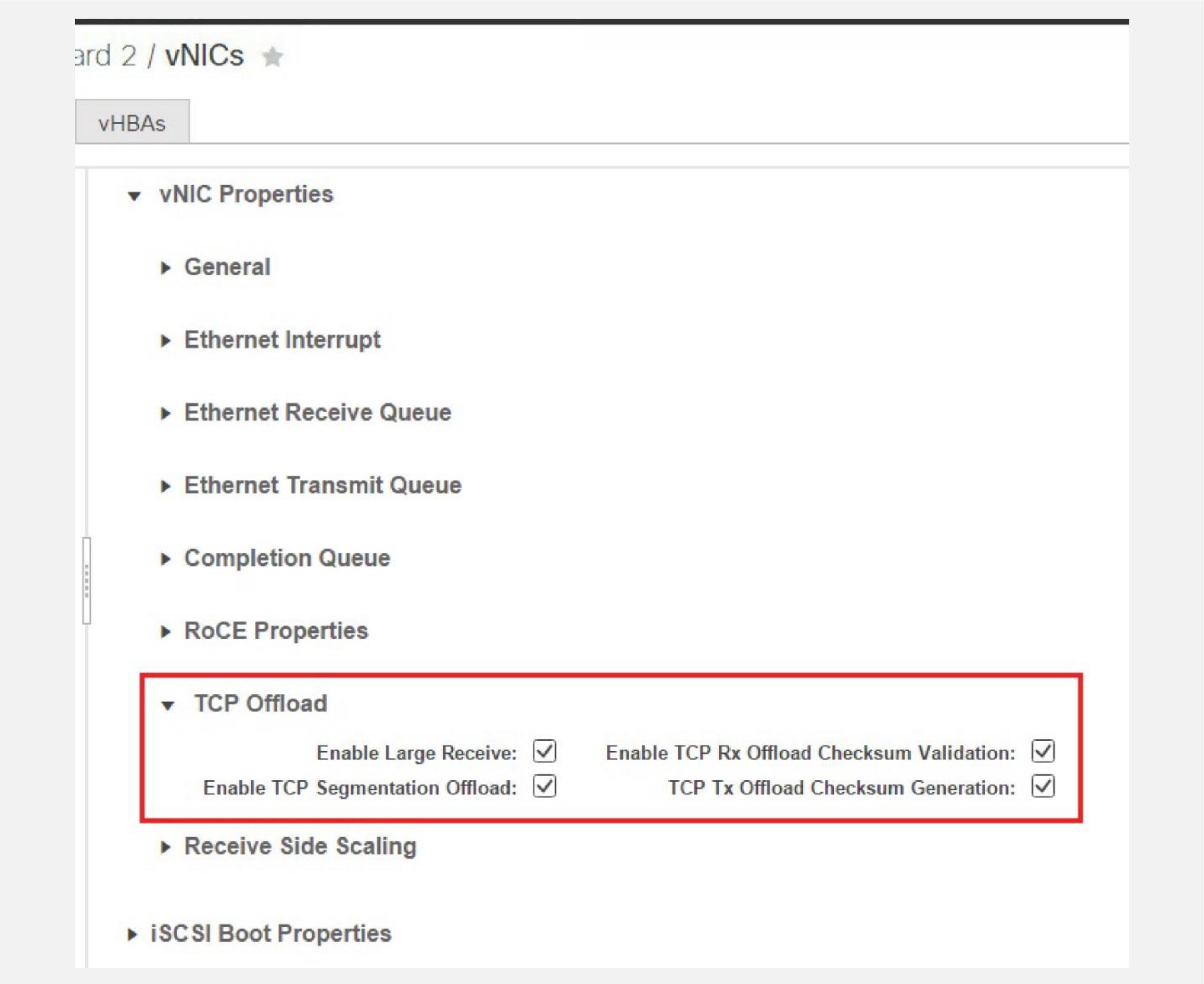
Network Virtualization using Generic Routing Encapsulation : ● Disabled ○ Enabled

Virtual Extensible LAN : ● Disabled ○ Enabled

Failback Timeout (Seconds) : 5 **[0-600]**

Interrupt Mode : ● MSI X ○ MSI ○ IN Tx

Figure 19. Setting offloads with Cisco UCS Manager



ard 2 / vNICs

vHBAs

▾ vNIC Properties

▸ General

▸ Ethernet Interrupt

▸ Ethernet Receive Queue

▸ Ethernet Transmit Queue

▸ Completion Queue

▸ RoCE Properties

▾ TCP Offload

Enable Large Receive: ☑ Enable TCP Rx Offload Checksum Validation: ☑

Enable TCP Segmentation Offload: ☑ TCP Tx Offload Checksum Generation: ☑

▸ Receive Side Scaling

▸ iSCSI Boot Properties

Figure 20. Setting offloads with Cisco IMC

Network overlay offloads

An overlay network provides agility and flexibility to system administrators, allowing them to connect both virtual and physical devices over an existing physical network. Many standards have evolved to provide this type of solution, including Virtual Extensible LAN (VXLAN), Network Virtualization using Generic Routing Encapsulation (NVGRE) and Generic Networking Virtualization Encapsulation (GENEVE). All these network overlay technologies use a tunneling technique whereby the original frame is encapsulated in an IP packet along with an extra header inserted into the overall packet. Figure 21 shows the VXLAN frame, Figure 22 shows the NVGRE frame, and Figure 23 shows the GENEVE frame. This additional header typically breaks existing offload engines, shifting the work back to the server and causing higher CPU utilization for the additional packet processing.

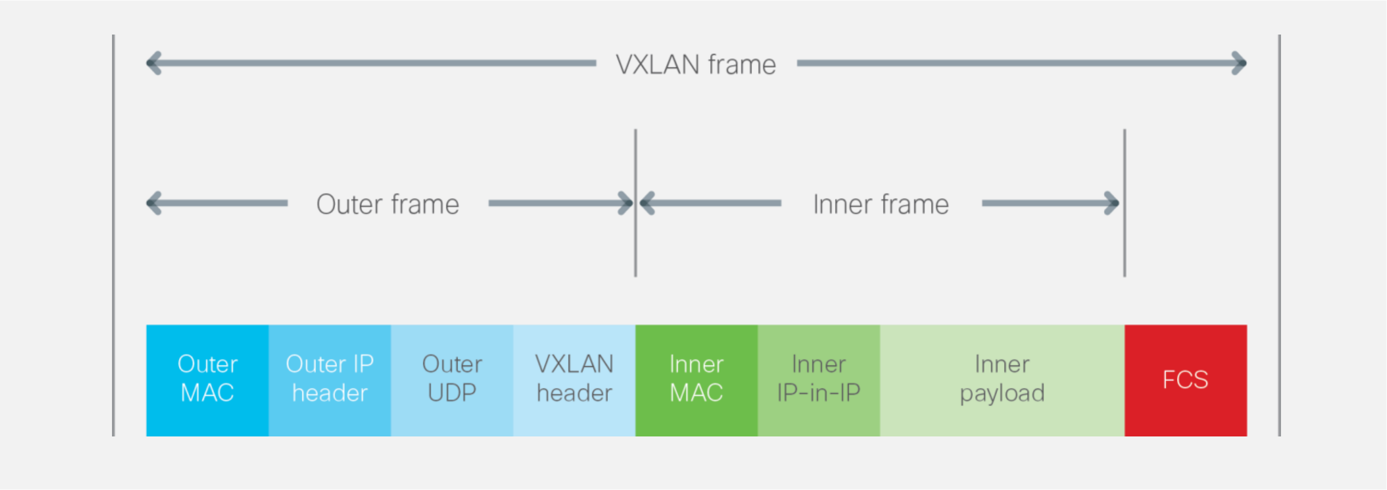


Figure 21. A VXLAN frame

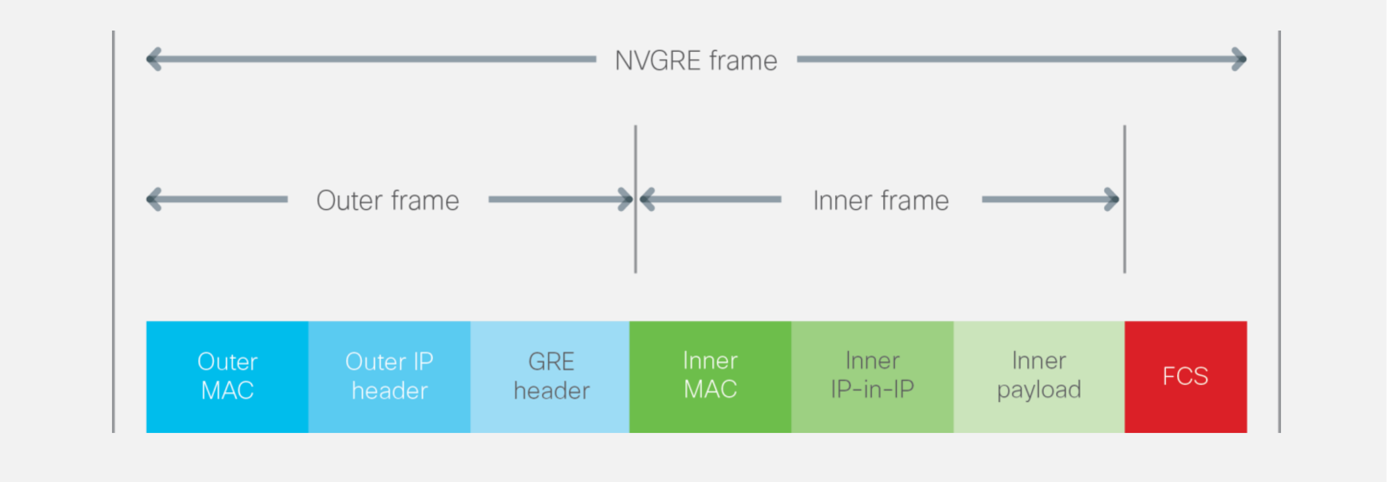


Figure 22. An NVGRE frame

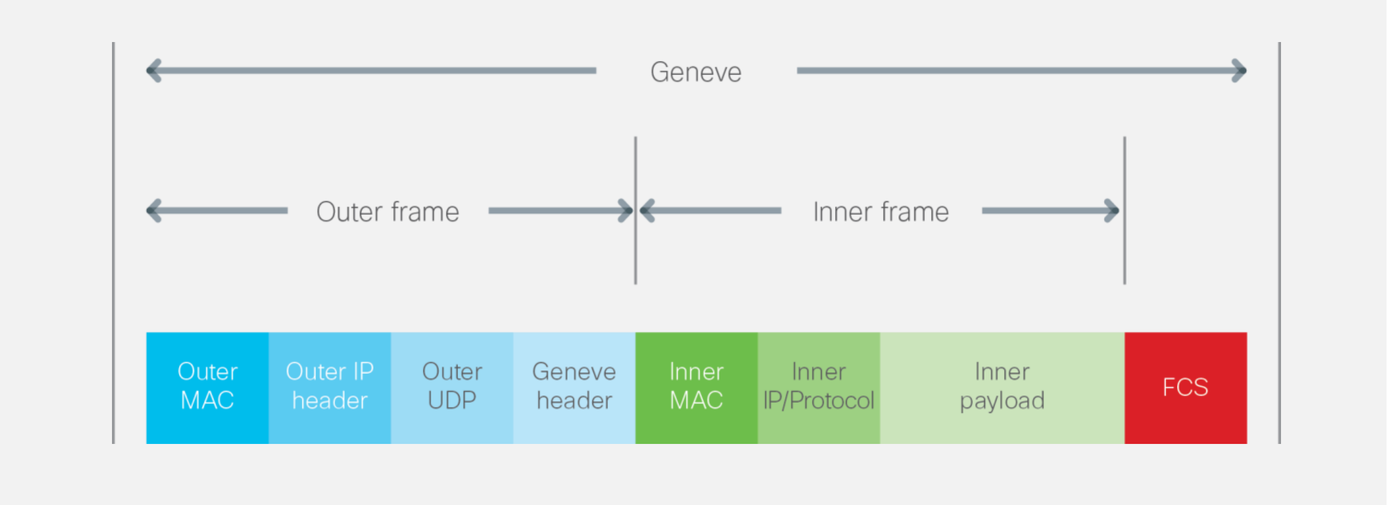


Figure 23. A Geneve frame

Network overlay offload is supported with the third and fourth generations of the Cisco UCS VIC (1300 and 1400 series). The overlay offload capabilities of VIC as of Cisco 4.1(3) UCS Manager and IMC releases are as follows.

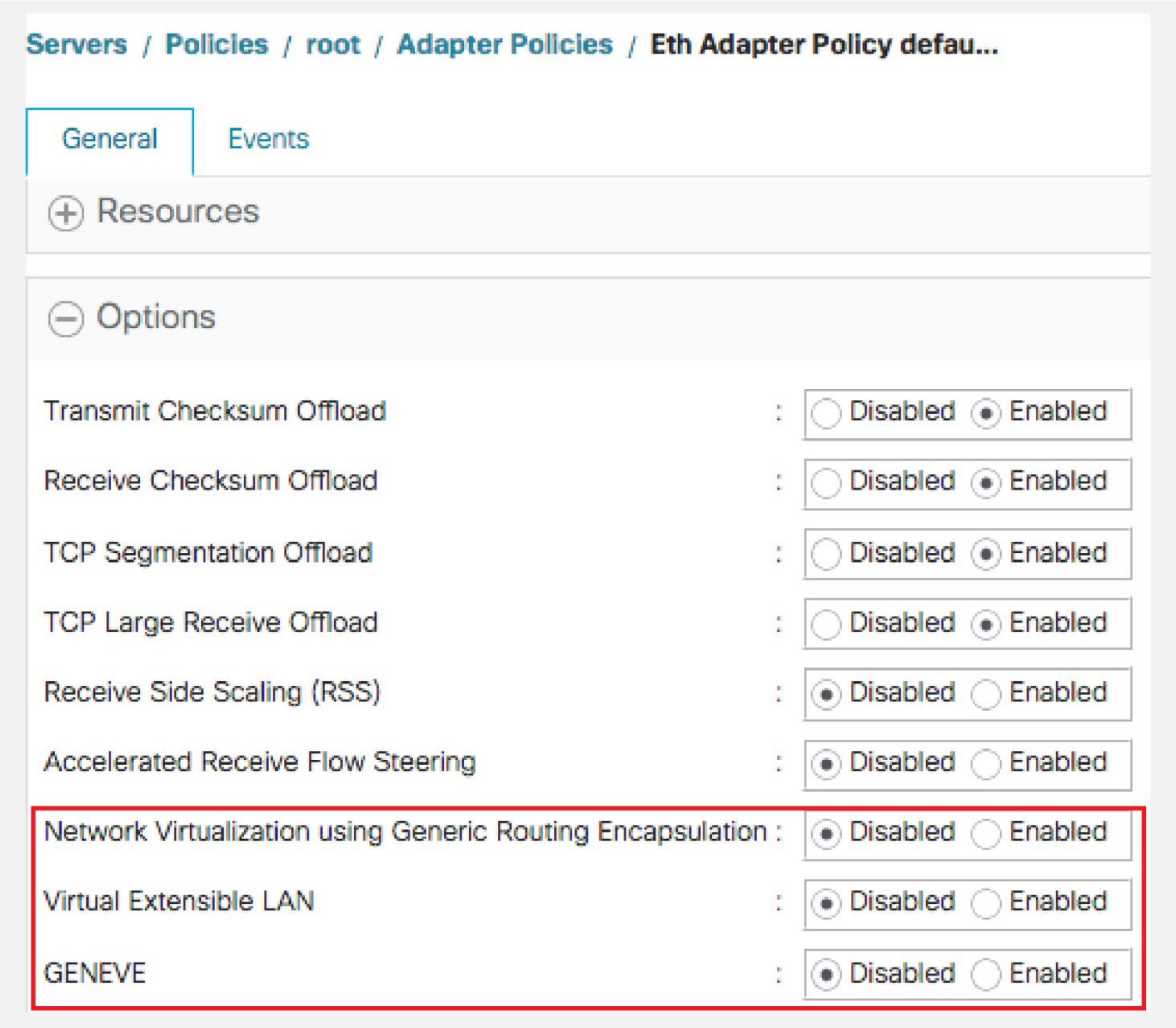
- VIC 1300 supports VXLAN and NVGRE offloads
- VIC 1400 support VXLAN, NVGRE and GENEVE offloads
- VXLAN offload is supported for VMware ESXi, Linux and Microsoft Windows
- NVGRE offload is supported for Microsoft Windows
- GENEVE offload is supported for VMware ESXi.

Additionally, note that the Cisco UCS VIC 1300 Series supports VXLAN offload with only IPv4 payload while the 1400 series doesn't have any such limitation and the Cisco UCS VIC 1400 supports VXLAN offloads with both IPv4 and IPv6 payloads. For some offload functions, the Cisco UCS VIC can process both the outer and inner headers, which reduces the overall burden on the CPU (Table 6).

Table 6. Network overlay support with offloads

Egress (transmit)	Outer frame	Inner frame
IP checksum	Yes	Yes
TCP and User Datagram Protocol	UDP only	Yes (for both protocols)
TSO	–	Yes
Ingress (I)	Outer frame	Inner frame
IP checksum	Yes	Yes
TCP and UDP checksum	UDP only	Yes (for both protocols)

Figures 24 and 25 show how to enable network overlay offloads using Cisco UCS Manager and Cisco IMC respectively.



Servers / Policies / root / Adapter Policies / Eth Adapter Policy defau...

Transmit Checksum Offload : ○ Disabled ● Enabled

Receive Checksum Offload : ○ Disabled ● Enabled

TCP Segmentation Offload : ○ Disabled ● Enabled

TCP Large Receive Offload : ○ Disabled ● Enabled

Receive Side Scaling (RSS) : ● Disabled ○ Enabled

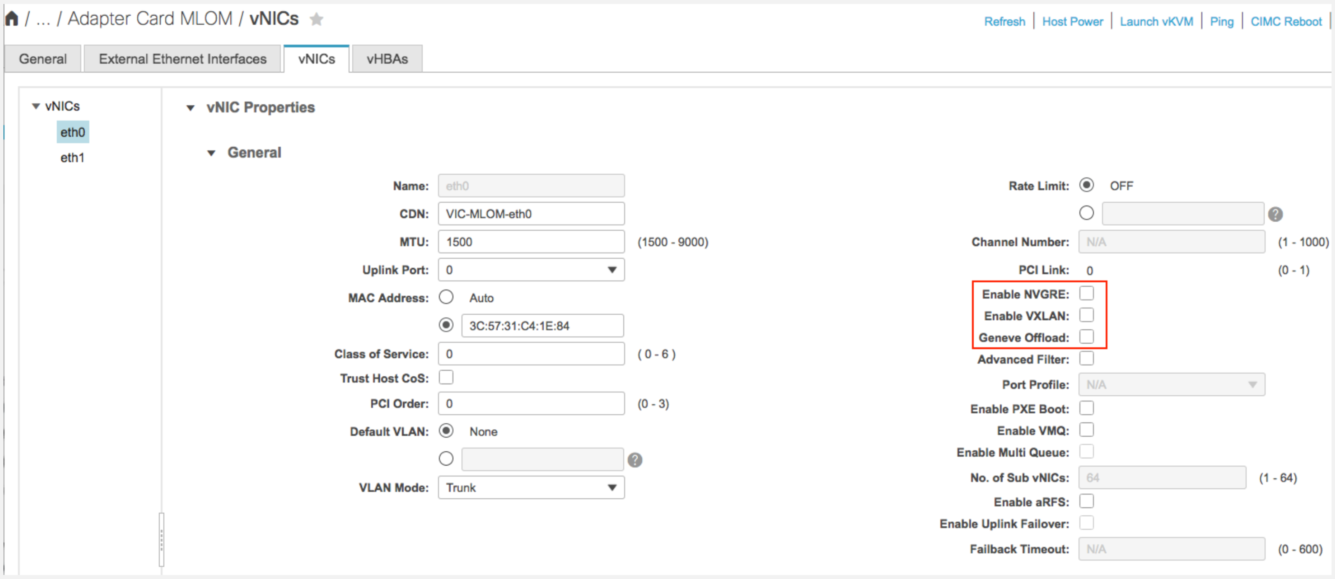
Accelerated Receive Flow Steering : ● Disabled ○ Enabled

Network Virtualization using Generic Routing Encapsulation : ● Disabled ○ Enabled

Virtual Extensible LAN : ● Disabled ○ Enabled

GENEVE : ● Disabled ○ Enabled

Figure 24. Cisco UCS Manager enablement for either VXLAN, NVGRE or GENEVE


/ ... / Adapter Card MLOM / vNICs
Refresh | Host Power | Launch vKVM | Ping | CIMC Reboot
General
External Ethernet Interfaces
vNICs
vHBAs
vNICs
eth0
eth1
vNIC Properties
General

Name: eth0
CDN: VIC-MLOM-eth0
MTU: 1500 (1500 - 9000)
Uplink Port: 0
MAC Address: ○ Auto ● 3C:57:31:C4:1E:84
Class of Service: 0 (0 - 6)
Trust Host CoS: □
PCI Order: 0 (0 - 3)
Default VLAN: ● None ○
VLAN Mode: Trunk

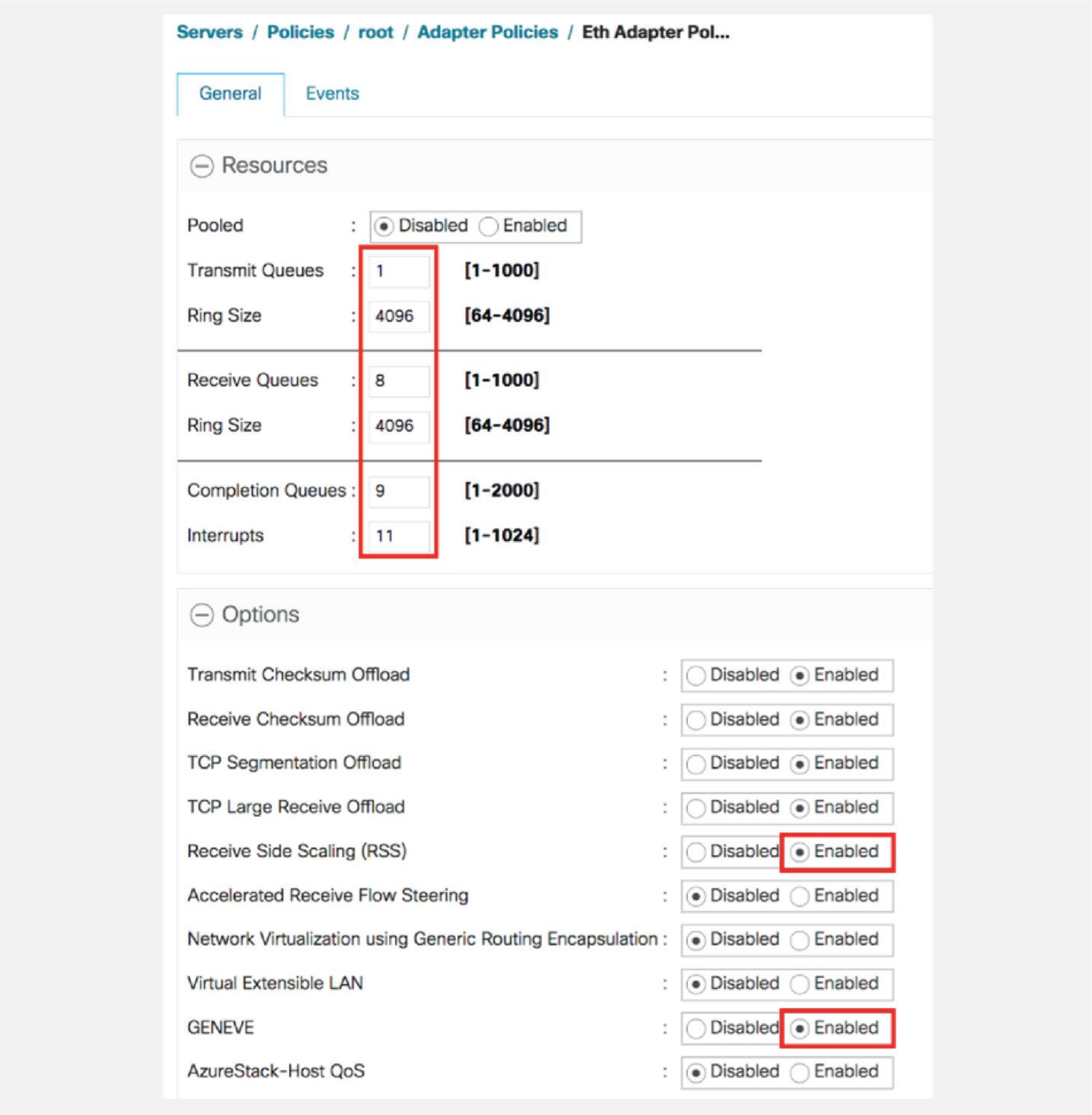
Rate Limit: ● OFF ○
Channel Number: N/A (1 - 1000)
PCI Link: 0 (0 - 1)
Enable NVGRE: □
Enable VXLAN: □
Geneve Offload: □
Advanced Filter: □
Port Profile: N/A
Enable PXE Boot: □
Enable VMQ: □
Enable Multi Queue: □
No. of Sub vNICs: 64 (1 - 64)
Enable aRFS: □
Enable Uplink Failover: □
Failback Timeout: N/A (0 - 600)

Note: You cannot enable the network overlay offload feature with Cisco NetFlow, user-space NIC (usNIC), or VMware NetQueue. Refer to the configuration guide and release notes for additional information. Cisco recommends enabling RSS with VXLAN offloads for greater network performance improvement for the server.

Figure 25. Cisco IMC enablement for NVGRE, VXLAN or GENEVE

VMware NSX-T uses GENEVE tunnels, and GENEVE offload can improve its networking performance. GENEVE offload support for the VMware ESXi hypervisor with 1400 series VICs has two possible solutions depending on whether NSX-T vSwitch (N-VDS or VDS 7) is using the standard mode or Enhanced Datapath mode. The VIC has two different drivers (nenic or nenic-ens), one for each mode. ESXi automatically loads the appropriate VIC driver based on the NSX-T configuration and does not require any actions from the user.

The GENEVE offload for NSX-T vSwitch in standard mode is set up by having the user first enable multiple RX-queues with RSS and Geneve offload on the UCS adapter policy for a vNIC (Figure 26). Then, on the NSX-T manager, the user selects the standard mode for vSwitch. In this case the 1400 series VIC enables RSS on the inner packet, TSO for inner IPv4/v6 packets and TX/RX checksum offload for IPv4/v6 packets. This helps reduce the CPU utilization and in achieving higher throughput. This feature is supported from the Cisco UCS Manager and Cisco IMC release 4.1(2) and up.



Servers / Policies / root / Adapter Policies / Eth Adapter Pol...

⊖ Resources

Pooled : ● Disabled ○ Enabled

Transmit Queues : 1 [1-1000]

Ring Size : 4096 [64-4096]

Receive Queues : 8 [1-1000]

Ring Size : 4096 [64-4096]

Completion Queues : 9 [1-2000]

Interrupts : 11 [1-1024]

⊖ Options

Transmit Checksum Offload : ○ Disabled ● Enabled

Receive Checksum Offload : ○ Disabled ● Enabled

TCP Segmentation Offload : ○ Disabled ● Enabled

TCP Large Receive Offload : ○ Disabled ● Enabled

Receive Side Scaling (RSS) : ○ Disabled ● Enabled

Accelerated Receive Flow Steering : ● Disabled ○ Enabled

Network Virtualization using Generic Routing Encapsulation : ● Disabled ○ Enabled

Virtual Extensible LAN : ● Disabled ○ Enabled

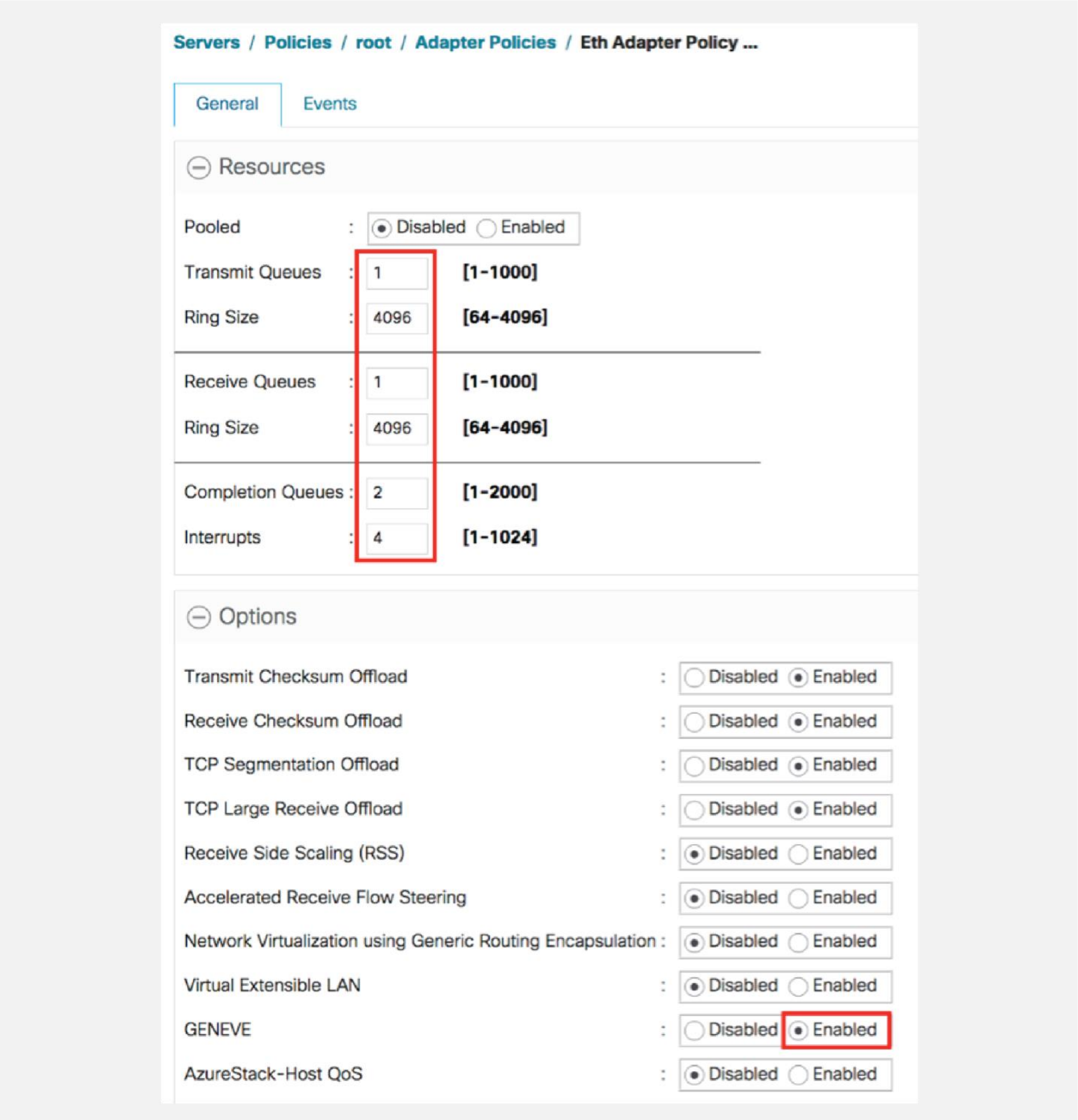
GENEVE : ○ Disabled ● Enabled

AzureStack-Host QoS : ● Disabled ○ Enabled

Figure 26. Adapter policy for GENEVE offload with RSS for NSX-T vSwitch in standard mode

Enhanced Network Stack (ENS) or alternatively Enhanced Datapath is the other option that can be turned on the VMware N-VDS or VDS 7 vSwitch. ENS borrows many techniques from the data plane development kit (DPDK) to achieve higher performance. Most notably, ENS utilizes polling to achieve high packet rates. When ENS is enabled, vSwitch uses the VMware ENS stack and utilizes the VIC ENS driver for its uplink ports.

To use ENS, the user first enables Geneve offload on the UCS adapter policy (Figure 27). Then, on the NSX-T manager, the user enables Enhanced Datapath mode on N-VDS or VDS 7. The 1400 series VIC with ENS uses single TX/RX queue, and there is no RSS support in this case. This feature is supported from Cisco UCS Manager and Cisco IMC release 4.1(3) and up.


Servers / Policies / root / Adapter Policies / Eth Adapter Policy ...
General
Events

Resources

Pooled : ● Disabled ○ Enabled

Transmit Queues : 1 [1-1000]

Ring Size : 4096 [64-4096]

Receive Queues : 1 [1-1000]

Ring Size : 4096 [64-4096]

Completion Queues : 2 [1-2000]

Interrupts : 4 [1-1024]

Options

Transmit Checksum Offload : ○ Disabled ● Enabled

Receive Checksum Offload : ○ Disabled ● Enabled

TCP Segmentation Offload : ○ Disabled ● Enabled

TCP Large Receive Offload : ○ Disabled ● Enabled

Receive Side Scaling (RSS) : ● Disabled ○ Enabled

Accelerated Receive Flow Steering : ● Disabled ○ Enabled

Network Virtualization using Generic Routing Encapsulation : ● Disabled ○ Enabled

Virtual Extensible LAN : ● Disabled ○ Enabled

GENEVE : ○ Disabled ● Enabled

AzureStack-Host QoS : ● Disabled ○ Enabled

Figure 27. Adapter policy for Geneve offload with ENS

Virtualization performance improvements

Server virtualization can provide significant benefits, allowing administrators to increase hardware utilization and more rapidly deploy applications. However, virtualization also introduces a new set of challenges, including slower network I/O performance for guest virtual machines than for bare-metal servers. The hypervisor is typically the bottleneck.

When the guest virtual machines run behind the hypervisor, all the I/O is serviced from a single transmit and receive queue, creating a bottleneck and burdening the hypervisor. The single transmit and receive queue pair limits the overall bandwidth that can flow to the guest machine because the bandwidth is being shared across all the guest machines. Moreover, the hypervisor must sort and switch the I/O traffic to the correct virtual machine.

To help reduce the burden for the hypervisor and eliminate the queue sharing, the Cisco UCS VIC can create dedicated queue pairs for guest machines running under both Microsoft Windows Hyper-V and VMware ESXi. This approach provides significant benefits. First, the transmit and receive queue pairs are no longer shared with other guest machines. Second, the hypervisor is no longer responsible for sorting and switching packets because packet steering is moved to the adapter. The adapter performs packet steering based on Layer 2 information such as MAC address and VLAN. As a result, the hypervisor is responsible only for moving the traffic between the adapter and the virtual machine. This approach improves I/O performance and frees the hypervisor for other tasks.

Note: This virtualization function is supported with the second-, third-, and fourth-generation Cisco UCS VIC 1200, 1300, and 1400 series.

The feature support across different generations of Cisco UCS VICs are as follows

- VIC 1200, 1300 and 1400 series support NetQueue for ESXi
- VIC 1300 and 1400 support virtual machine queue (VMQ) for Windows Hyper-V
- VIC 1400 support VMMQ and is recommended over VMQ for Windows Hyper-V

The Cisco UCS VIC can support up to 128 virtual machine queues (VMQs) per vNIC and a total of 256 VMQs per adapter. While the VIC hardware is capable of supporting these numbers, there are operating system specific limits. Included below are the limits per vNIC for various operating systems.

- VMware ESX 6.0 and lower: 8 VMQ
- VMware ESXi 6.5 and up: 16 VMQ
- Microsoft Windows Server Hyper-V: 64 VMQ or 64 VMMQ vPorts

Virtual machine multi-queue (VMMQ) is an enhancement to VMQ and the Cisco UCS VIC’s RSS capabilities and this is supported on the VIC 1400 series for Windows Server 2016 and up.

In Cisco UCS Manager, VMQ connection policy is the common configuration utility that is used to define NetQueue for ESXi and VMQ or VMMQ for Windows Hyper-V. Note that in Cisco UCS Manager and IMC, the NetQueue for ESXi is referenced as VMQ, note this when enabling this feature to avoid confusion.

Enabling these features require configuration on both the Cisco UCS VIC and the operating system, as described in the sections that follow.

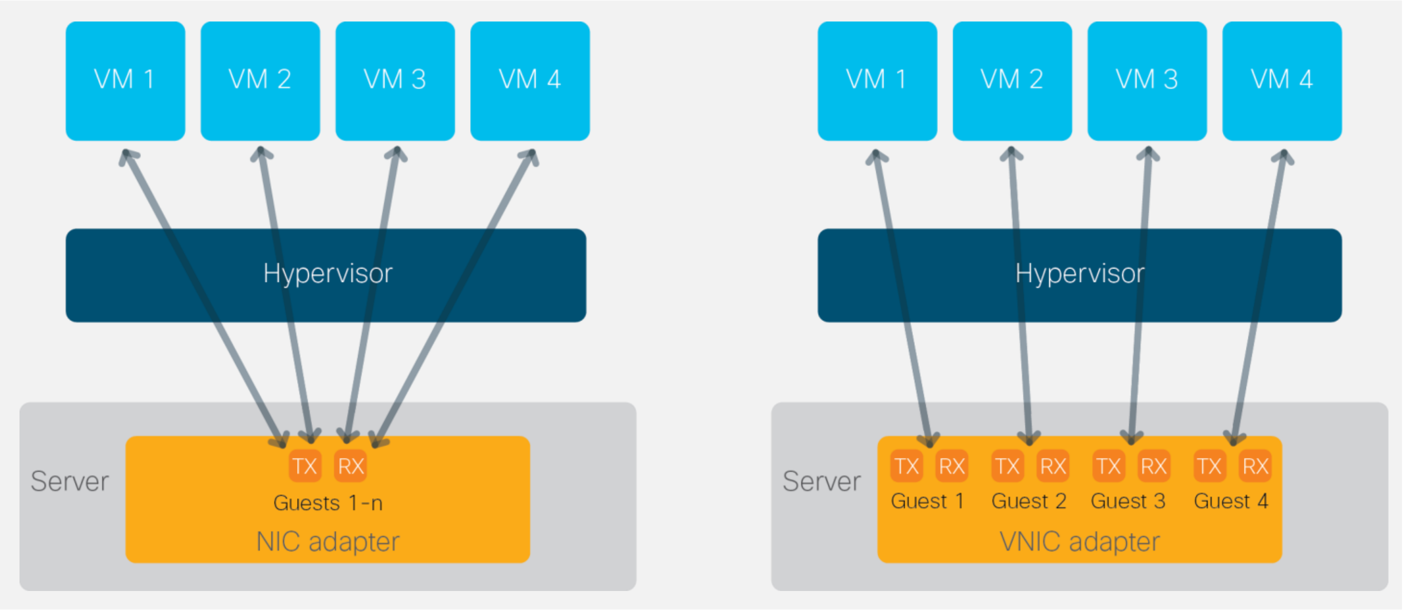


Figure 28. The Cisco UCS VIC provides multiple transmit and receive queues to service individual vNICs

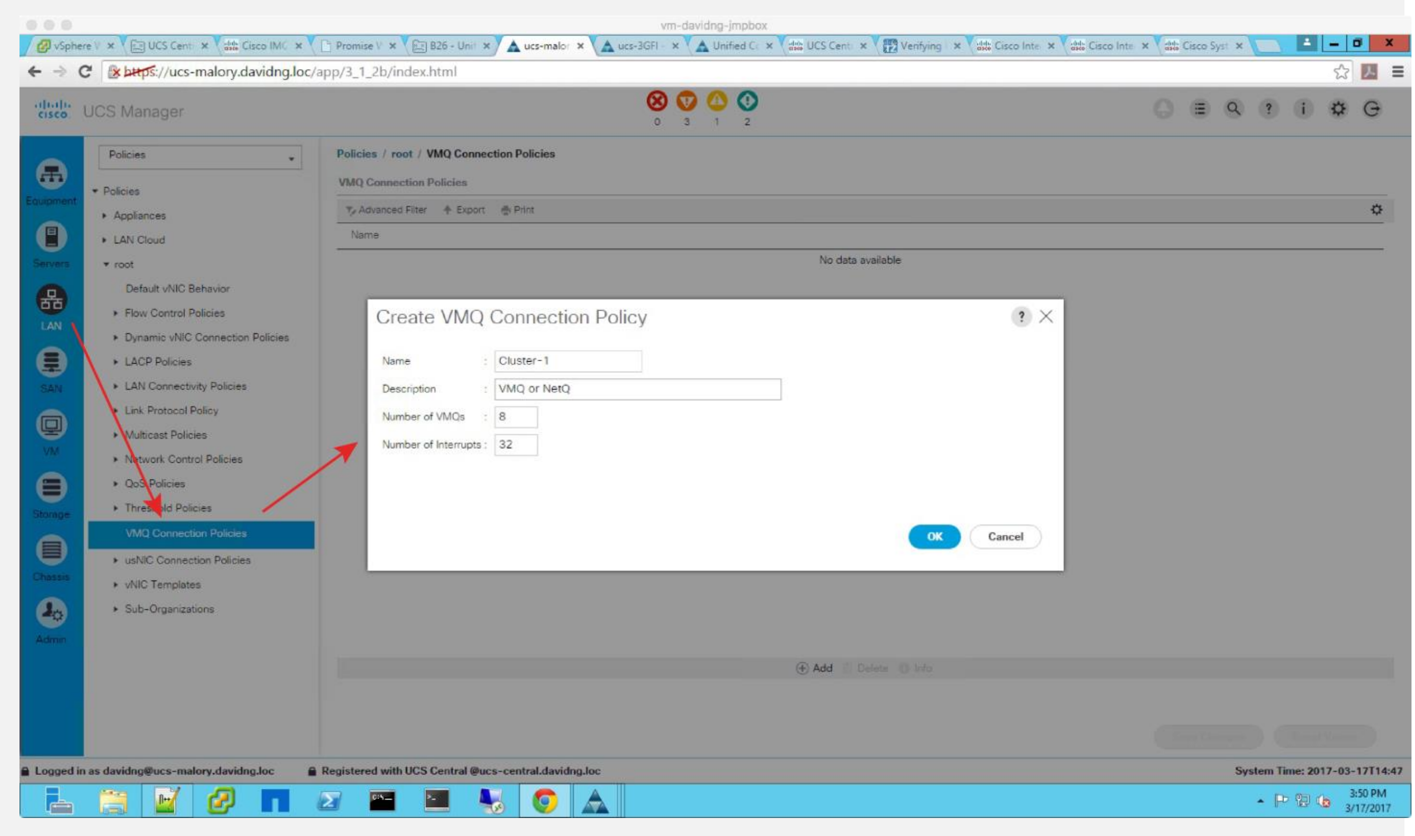
Configuring VMQ connection policy with Cisco UCS Manager

Cisco UCS Manager uses the same configuration method for both Windows Hyper-V and VMware ESXi. The configuration is implemented through the VMQ connection policy and is bound to an individual vNIC profile. The connection policy is located on the LAN tab (Figure 29). Both the number of hardware queues (VMQs) and the number of interrupts can be configured. The number of VMQs essentially equals the number of guest machines being configured. Interrupts are typically set as “2 x VMQ + 2” for VMware ESXi.

For Microsoft Windows Server, VMQ is supported for 1200 and 1300 series VICs and the configurations are done from the VMQ connection policy with the number of interrupts being “2 x VMQ + 2” rounded up to the next power of 2 up to a max of 128 (Table 7). For 1400 series VICs, VMMQ is the recommended configuration and its defined by specifying the MQ policy in the VMQ connection policy.

Table 7. Sample VMQ connection policy settings

Parameter	4 virtual machines		8 virtual machines	
	Microsoft Windows	VMware ESXi	Microsoft Windows	VMware ESXi
VMQs	4	4	8	8
Interrupts	16	10	32	18


https://ucs-malory.davidng.loc/app/3_1_2b/index.html
UCS Manager
Policies
Appliances
LAN Cloud
root
Default vNIC Behavior
Flow Control Policies
Dynamic vNIC Connection Policies
LACP Policies
LAN Connectivity Policies
Link Protocol Policy
Multicast Policies
Network Control Policies
QoS Policies
Threshold Policies
VMQ Connection Policies
usNIC Connection Policies
vNIC Templates
Sub-Organizations
Policies / root / VMQ Connection Policies
No data available
OK
Cancel
Logged in as davidng@ucs-malory.davidng.loc
Registered with UCS Central @ucs-central.davidng.loc
System Time: 2017-03-17T14:47

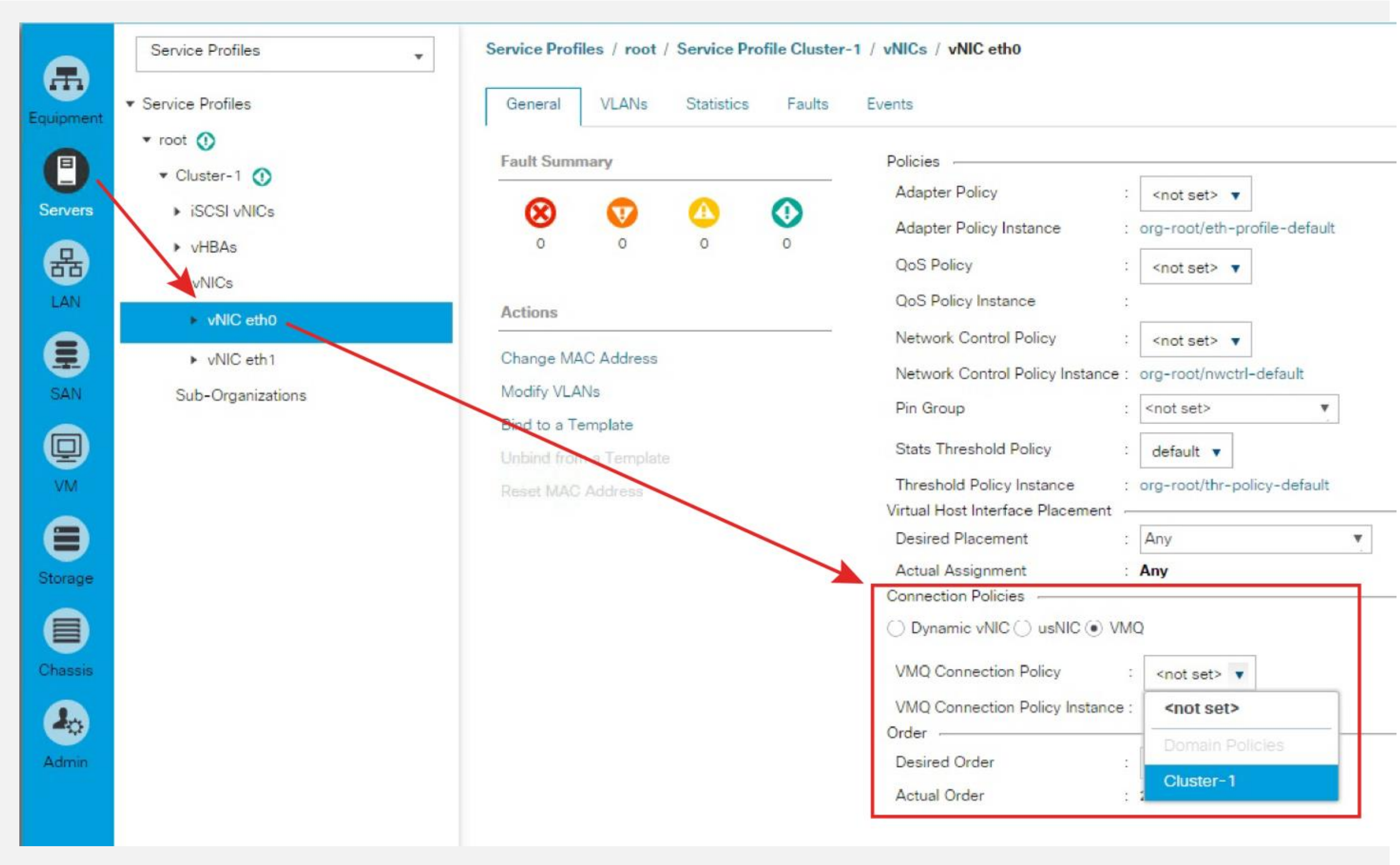
Create VMQ Connection Policy

Name : Cluster-1
Description : VMQ or NetQ
Number of VMQs : 8
Number of Interrupts : 32

Figure 29. Configuring a VMQ connection policy

The VMQ connection policy must then be bound to the individual vNIC profile (Figure 30).

Note: When the VMQ connection policy is set, it supersedes the adapter policy.



Policies

Adapter Policy	:	<not set>
Adapter Policy Instance	:	org-root/eth-profile-default
QoS Policy	:	<not set>
QoS Policy Instance	:	
Network Control Policy	:	<not set>
Network Control Policy Instance	:	org-root/nwctrl-default
Pin Group	:	<not set>
Stats Threshold Policy	:	default
Threshold Policy Instance	:	org-root/thr-policy-default

Virtual Host Interface Placement

Desired Placement	:	Any
Actual Assignment	:	**Any**

Connection Policies

○ Dynamic vNIC ○ usNIC ● VMQ

VMQ Connection Policy	:	<not set>
VMQ Connection Policy Instance	:	

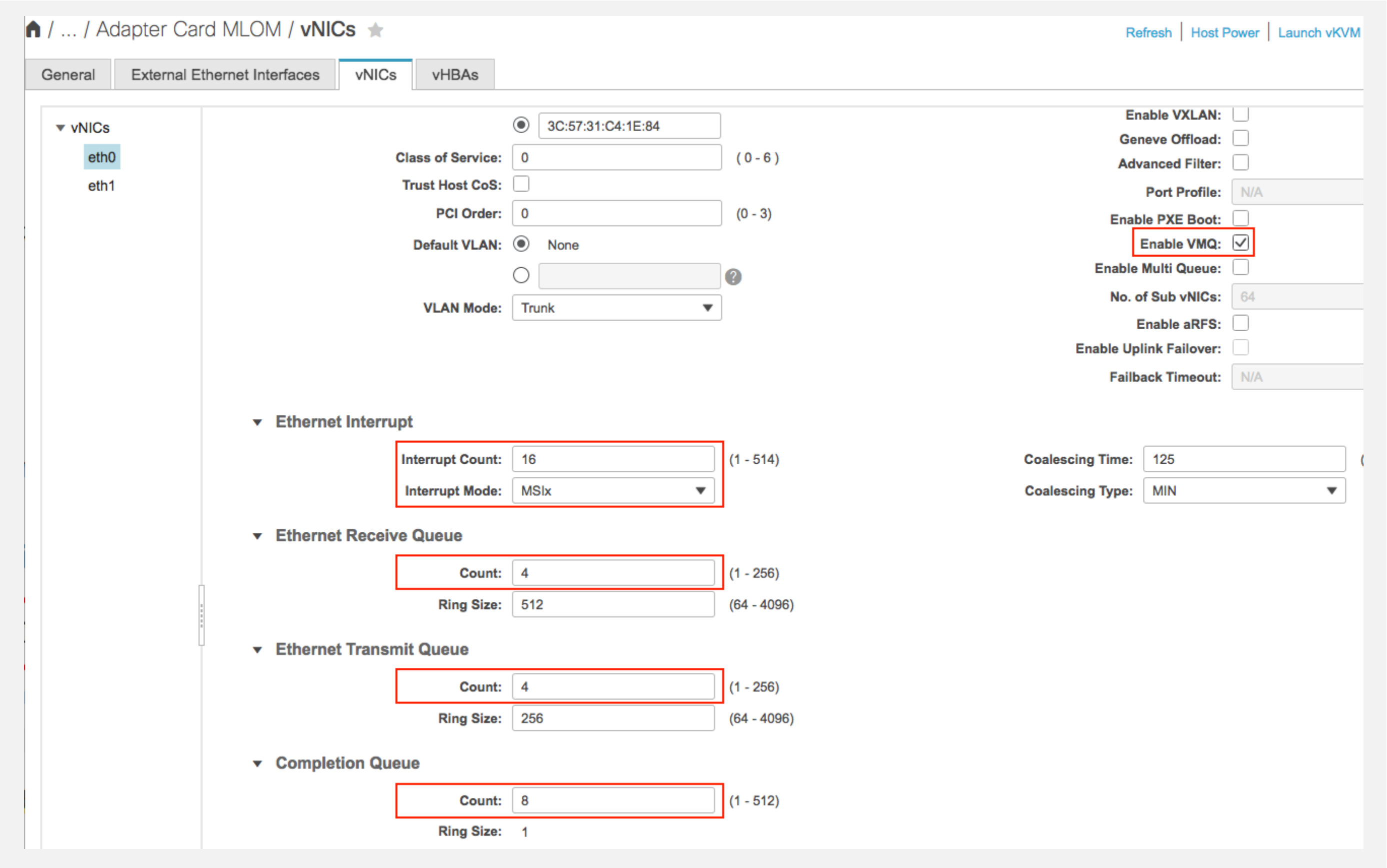
Order

Desired Order	:	
Actual Order	:	

Figure 30. VMQ connection policy bound to the vNIC profile supersedes the adapter policy

Configuring VMQ connection policy with Cisco IMC

Configuring multiple transmit and receive queues in the Cisco IMC is different than doing so in Cisco UCS Manager, because the controller has no concept of VMQ connection policy. Instead, you need to adjust the queues (TX, RX, and CQ) and interrupts accordingly. First, you need to enable the VMQ (Figure 31).


/ ... / Adapter Card MLOM / vNICs
Refresh | Host Power | Launch vKVM
General
External Ethernet Interfaces
vNICs
vHBAs
vNICs
eth0
eth1

● 3C:57:31:C4:1E:84

Class of Service: 0 (0 - 6)

Trust Host CoS: □

PCI Order: 0 (0 - 3)

Default VLAN: ● None ○

VLAN Mode: Trunk

Enable VXLAN: □

Geneve Offload: □

Advanced Filter: □

Port Profile: N/A

Enable PXE Boot: □

Enable VMQ: ☑

Enable Multi Queue: □

No. of Sub vNICs: 64

Enable aRFS: □

Enable Uplink Failover: □

Failback Timeout: N/A

Ethernet Interrupt

Interrupt Count: 16 (1 - 514)

Interrupt Mode: MSIx

Coalescing Time: 125

Coalescing Type: MIN

Ethernet Receive Queue

Count: 4 (1 - 256)

Ring Size: 512 (64 - 4096)

Ethernet Transmit Queue

Count: 4 (1 - 256)

Ring Size: 256 (64 - 4096)

Completion Queue

Count: 8 (1 - 512)

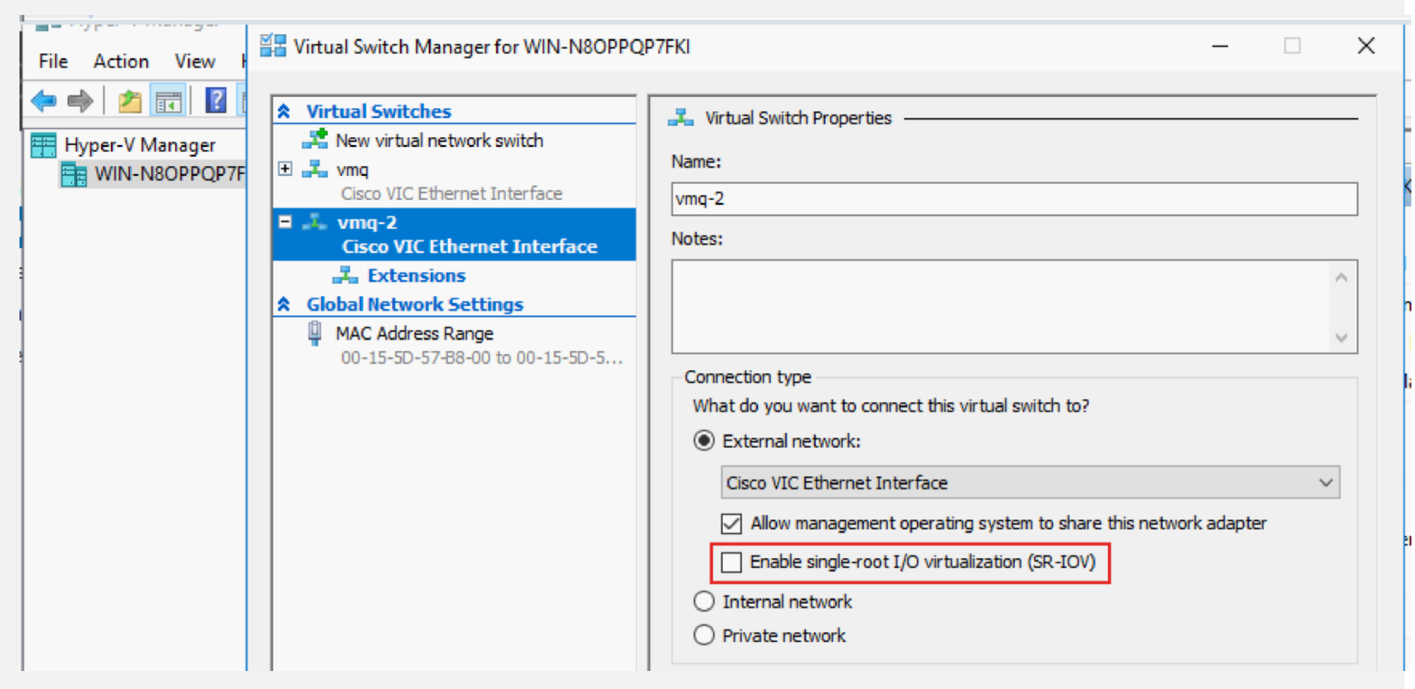
Ring Size: 1

Figure 31. VMQ can be enabled in both Microsoft Windows and VMware ESXi

The number of transmit queues, receive queues, and completion queues should be equal to the number of virtual machines being deployed. The interrupt calculation is the same as that shown earlier for Cisco UCS Manager. The interrupt mode should also be set to message-signaled interrupt X (MSIx).

Configuring VMQ on Microsoft Windows

Several settings need to be configured on the OS side to enable VMQ. First, disable single-root I/O virtualization (SR-IOV) on the virtual switch in the Virtual Switch Manager (Figure 32).


File
Action
View
Hyper-V Manager
WIN-N8OPPQP7F
Virtual Switch Manager for WIN-N8OPPQP7FKI
Virtual Switches
New virtual network switch
vmq
Cisco VIC Ethernet Interface
vmq-2
Cisco VIC Ethernet Interface
Extensions
Global Network Settings
MAC Address Range
00-15-5D-57-B8-00 to 00-15-5D-5...
Virtual Switch Properties

Name: vmq-2

Notes:

Connection type

What do you want to connect this virtual switch to?

● External network:

Cisco VIC Ethernet Interface

☑ Allow management operating system to share this network adapter

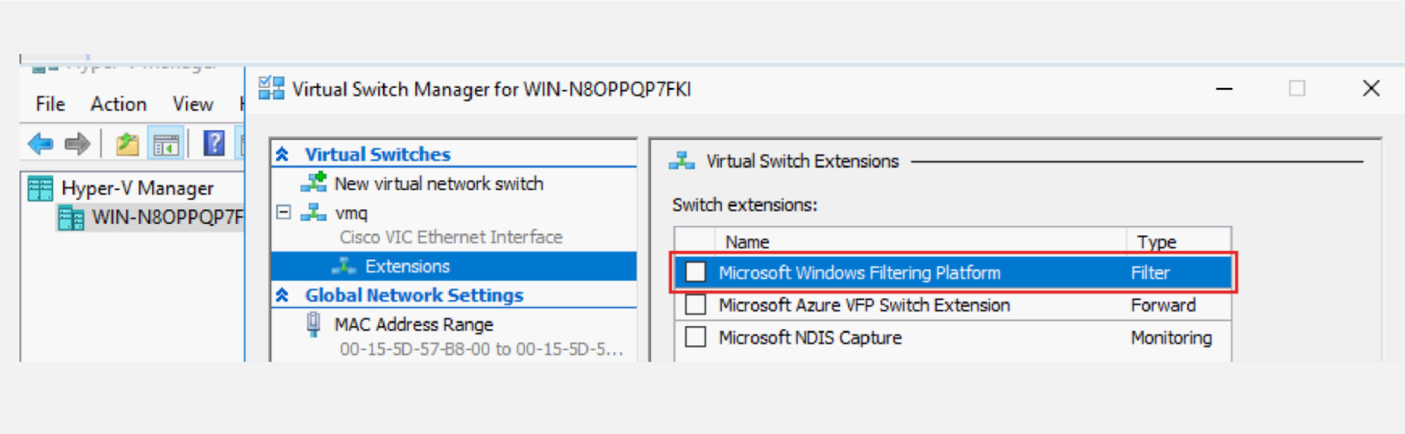
□ Enable single-root I/O virtualization (SR-IOV)

○ Internal network

○ Private network

Figure 32. Disable SR-IOV on the virtual switch

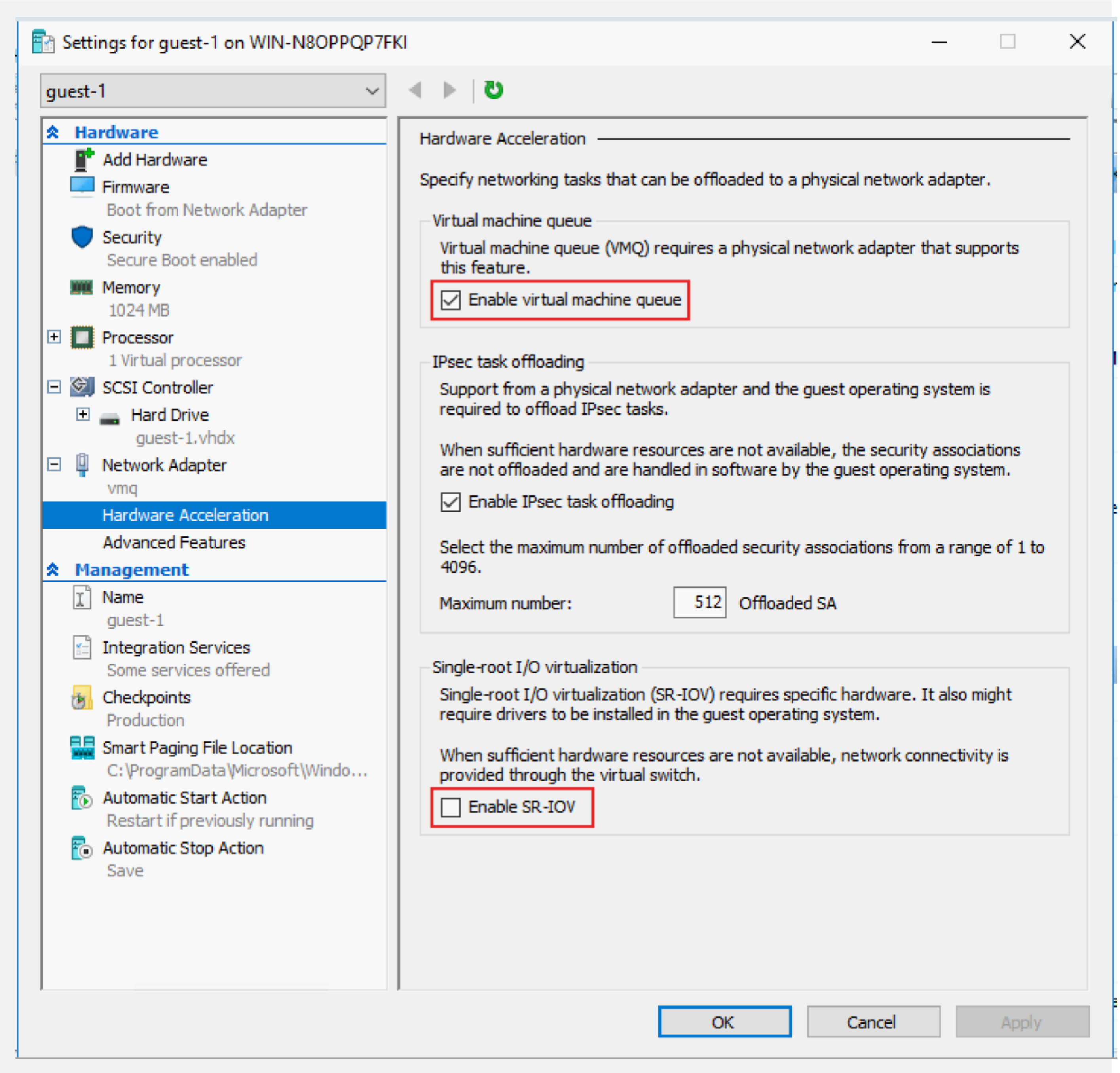
Next, for the extension for the interface, verify that the filter driver is not attached to the virtual switch (Figure 33).


File
Action
View
Hyper-V Manager
WIN-N8OPPQP7F
Virtual Switch Manager for WIN-N8OPPQP7FKI
Virtual Switches
New virtual network switch
vmq
Cisco VIC Ethernet Interface
Extensions
Global Network Settings
MAC Address Range
00-15-5D-57-B8-00 to 00-15-5D-5...
Virtual Switch Extensions

Switch extensions:

Name	Type
□ Microsoft Windows Filtering Platform	Filter
□ Microsoft Azure VFP Switch Extension	Forward
□ Microsoft NDIS Capture	Monitoring

Figure 33. Verify that no filter driver is attached to the virtual switch



Settings for guest-1 on WIN-N8OPPQP7FKI

guest-1

Hardware

- Add Hardware
- Firmware
  - Boot from Network Adapter
- Security
  - Secure Boot enabled
- Memory
  - 1024 MB
- Processor
  - 1 Virtual processor
- SCSI Controller
  - Hard Drive
    - guest-1.vhdx
- Network Adapter
  - vmq
  - Hardware Acceleration
  - Advanced Features

Management

- Name
  - guest-1
- Integration Services
  - Some services offered
- Checkpoints
  - Production
- Smart Paging File Location
  - C:\ProgramData\Microsoft\Windo...
- Automatic Start Action
  - Restart if previously running
- Automatic Stop Action
  - Save

Hardware Acceleration

Specify networking tasks that can be offloaded to a physical network adapter.

Virtual machine queue

Virtual machine queue (VMQ) requires a physical network adapter that supports this feature.

☑ Enable virtual machine queue

IPsec task offloading

Support from a physical network adapter and the guest operating system is required to offload IPsec tasks.

When sufficient hardware resources are not available, the security associations are not offloaded and are handled in software by the guest operating system.

☑ Enable IPsec task offloading

Select the maximum number of offloaded security associations from a range of 1 to 4096.

Maximum number: 512 Offloaded SA

Single-root I/O virtualization

Single-root I/O virtualization (SR-IOV) requires specific hardware. It also might require drivers to be installed in the guest operating system.

When sufficient hardware resources are not available, network connectivity is provided through the virtual switch.

□ Enable SR-IOV

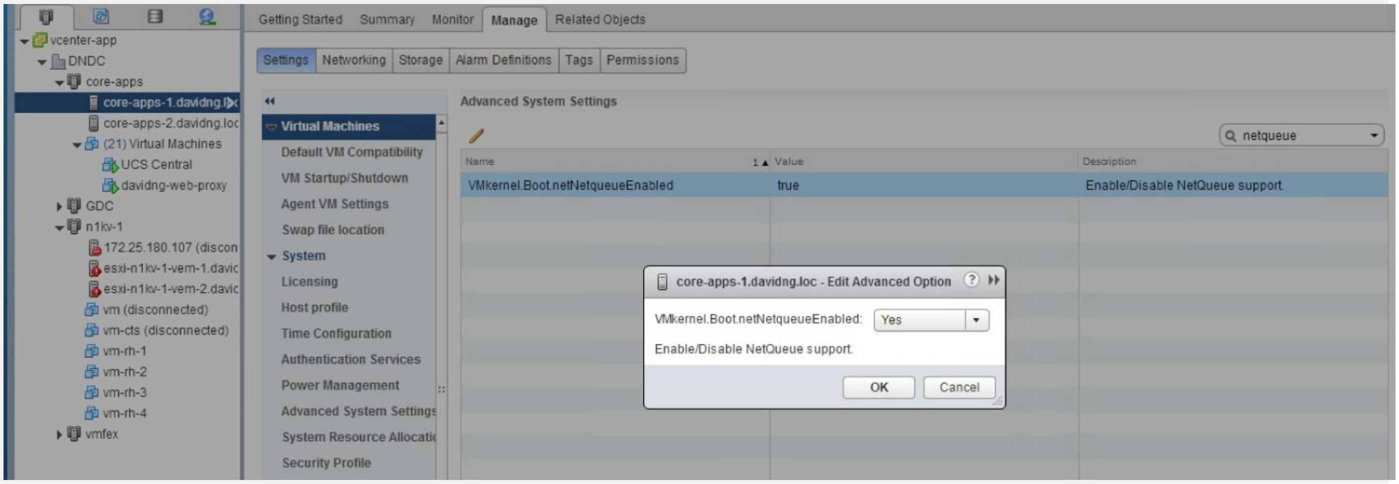
Figure 34. Guest virtual machine configuration

Finally, on the guest machine, check the virtual machine queue and make sure that SR-IOV is disabled (Figure 34).

Configuring VMQ on VMware ESXi (NetQueue)

For VMware ESXi, you need to consider some scale limits. The VMware ESXi eNIC driver limits the number of VMQs to 16 per vNIC for all ESXi versions above 6.0, while for ESXi 6.0 or lower the number of VMQs supported is limited to 8 per vNIC.

To configure VMQ on VMware ESXi, verify that VMkernel.Boot.netNetqueueEnabled is set to Yes (Figure 35).



Name	Value	Description
VMkernel.Boot.netNetqueueEnabled	true	Enable/Disable NetQueue support.

core-apps-1.davidng.loc - Edit Advanced Option

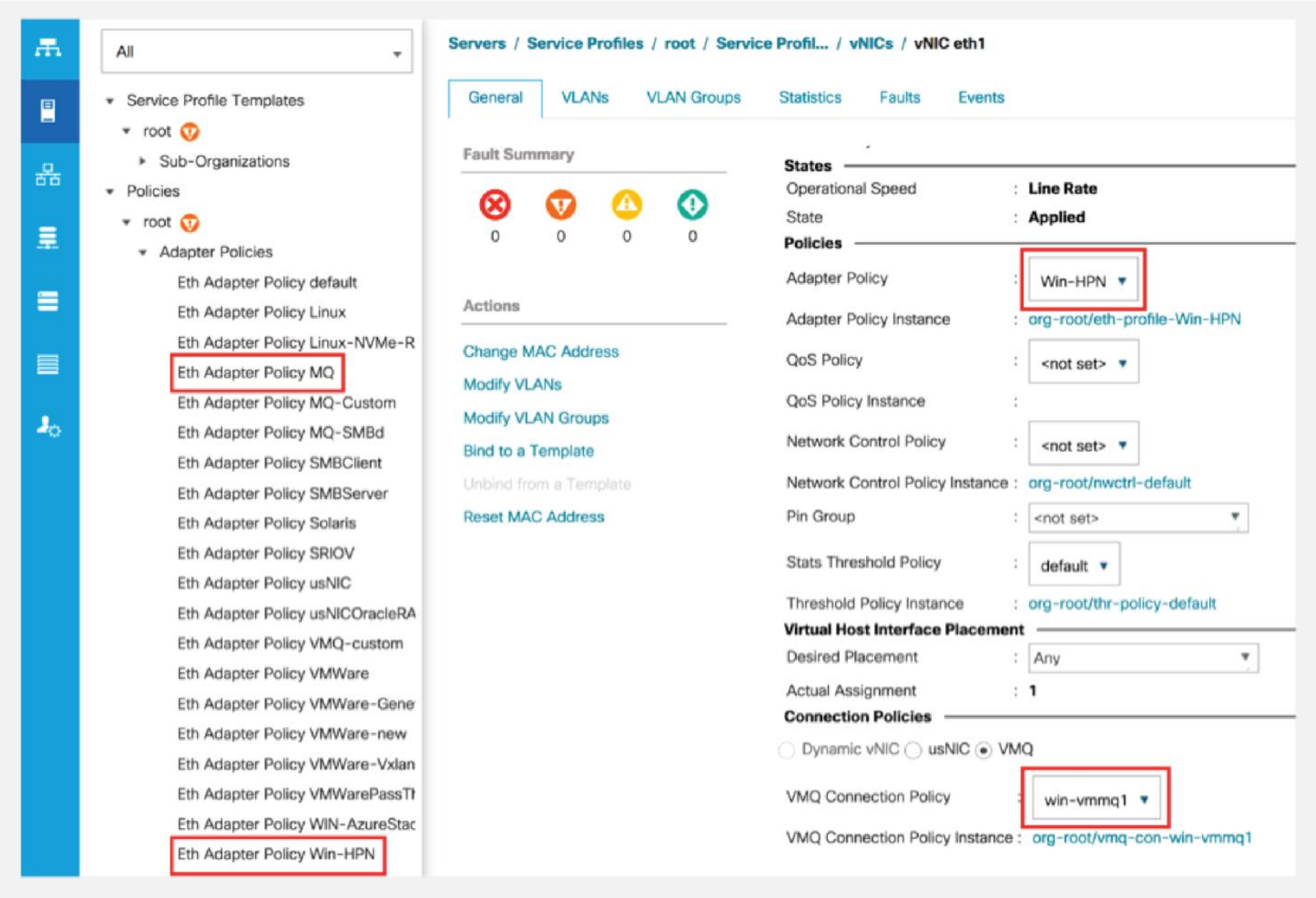
VMkernel.Boot.netNetqueueEnabled: Yes

Enable/Disable NetQueue support.

Figure 35. VMware ESXi configuration

Configuring VMMQ connection policy with Cisco UCS Manager

VMMQ is supported on 1400 series VICs from Microsoft Windows Server 2016 and up. The VMMQ configuration is implemented through the VMQ connection policy and bound to a vNIC. The VMQ connection policy is located on the LAN tab (Figure 36). For VMMQ, a VMQ connection policy specifies the number of sub-vNICs or vPorts in Hyper-V and also requires a multi-queue policy which specifies the RX and TX queues and the Interrupts required for the vPorts. For ease of configuration, Cisco UCS Manager has predefined adapter-policy (“Win-HPN”) and multi-queue policy (“MQ”) which can be used for enabling 64 vPorts per vNIC.


All
Service Profile Templates
root
Sub-Organizations
Policies
root
Adapter Policies
Eth Adapter Policy default
Eth Adapter Policy Linux
Eth Adapter Policy Linux-NVMe-R
Eth Adapter Policy MQ
Eth Adapter Policy MQ-Custom
Eth Adapter Policy MQ-SMBd
Eth Adapter Policy SMBClient
Eth Adapter Policy SMBServer
Eth Adapter Policy Solaris
Eth Adapter Policy SRIOV
Eth Adapter Policy usNIC
Eth Adapter Policy usNICOracleRA
Eth Adapter Policy VMQ-custom
Eth Adapter Policy VMWare
Eth Adapter Policy VMWare-Gene
Eth Adapter Policy VMWare-new
Eth Adapter Policy VMWare-Vxlan
Eth Adapter Policy VMWarePassTl
Eth Adapter Policy WIN-AzureStac
Eth Adapter Policy Win-HPN
Servers / Service Profiles / root / Service Profil... / vNICs / vNIC eth1
General
VLANs
VLAN Groups
Statistics
Faults
Events
Fault Summary
0
0
0
0
Actions
Change MAC Address
Modify VLANs
Modify VLAN Groups
Bind to a Template
Unbind from a Template
Reset MAC Address

States

Operational Speed : **Line Rate**

State : **Applied**

Policies

Adapter Policy : Win-HPN

Adapter Policy Instance : org-root/eth-profile-Win-HPN

QoS Policy : <not set>

QoS Policy Instance :

Network Control Policy : <not set>

Network Control Policy Instance : org-root/nwctrl-default

Pin Group : <not set>

Stats Threshold Policy : default

Threshold Policy Instance : org-root/thr-policy-default

Virtual Host Interface Placement

Desired Placement : Any

Actual Assignment : **1**

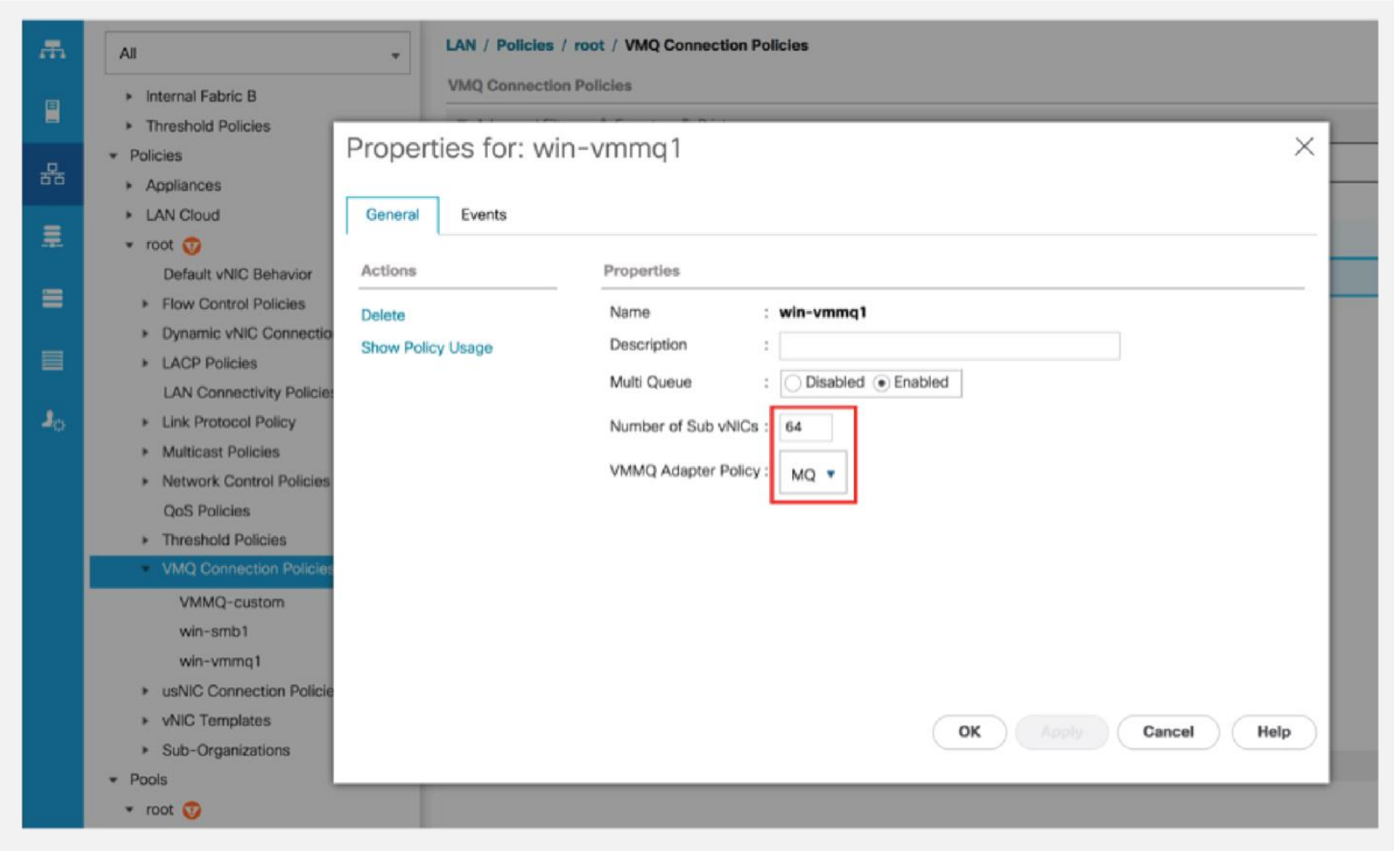
Connection Policies

○ Dynamic vNIC ○ usNIC ● VMQ

VMQ Connection Policy : win-vmmq1

VMQ Connection Policy Instance : org-root/vmq-con-win-vmmq1

Figure 36. VMMQ configuration with Cisco UCS Manager



Properties for: win-vmmq1

Name : win-vmmq1

Description :

Multi Queue : ○ Disabled ● Enabled

Number of Sub vNICs : 64

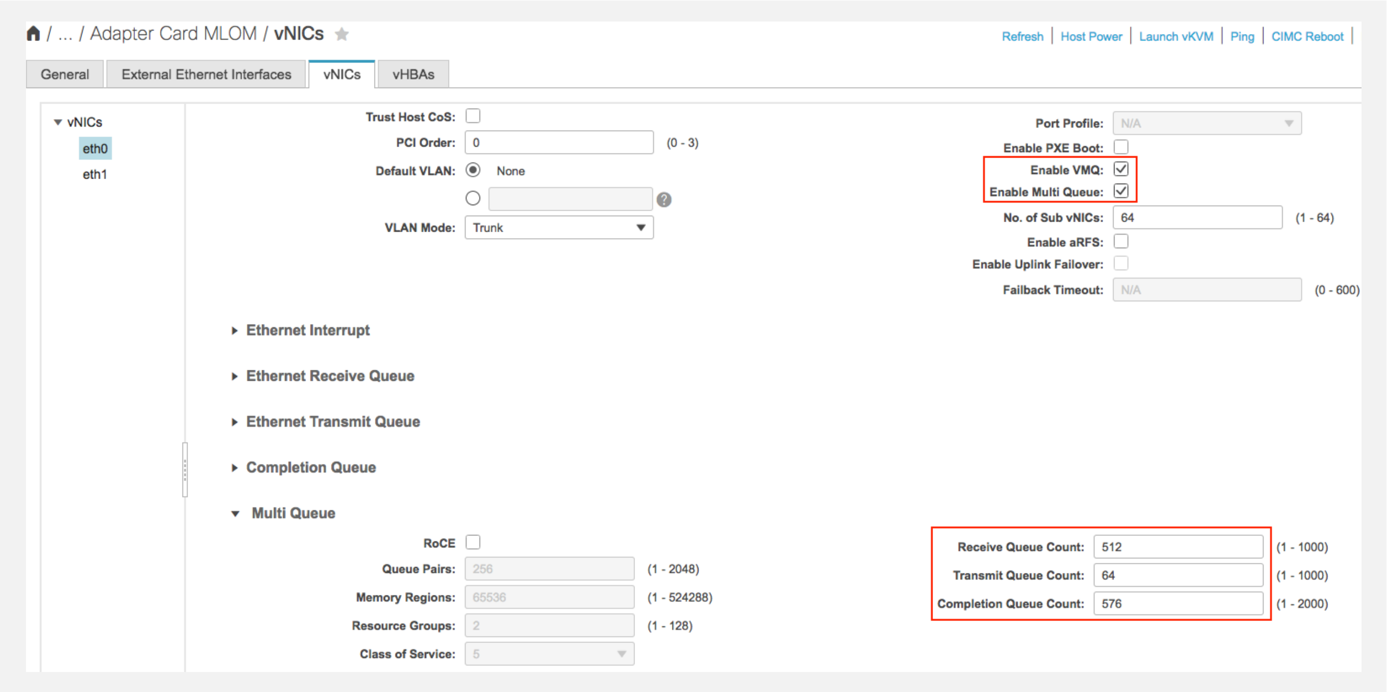
VMMQ Adapter Policy : MQ

Figure 37. Multi Queue configuration

VMMQ is also used for Microsoft Windows ROCEv2 mode 2 support on 1400 series VICs. Refer the UCS ROCE configuration guide for more details. For ROCEv2 on Microsoft Windows, use the predefined Cisco UCS Manager adapter-policy “Win-HPN-SMB” and the multi-queue policy “MQ-SMB” in the VMQ connection policy for the vNIC.

Configuring VMMQ connection policy with Cisco IMC

Configuring VMMQ on Cisco IMC involves enabling VMQ on the vNIC and configuring the multi-queue as shown below. The default configuration for the multi-queue is good for 64 vPorts or sub vNICs.



Trust Host CoS: ☐
PCI Order: 0 (0 - 3)
Default VLAN: ● None
○
VLAN Mode: Trunk

Port Profile: N/A
Enable PXE Boot: ☐
Enable VMQ: ☑
Enable Multi Queue: ☑
No. of Sub vNICs: 64 (1 - 64)
Enable aRFS: ☐
Enable Uplink Failover: ☐
Failback Timeout: N/A (0 - 600)

RoCE ☐
Queue Pairs: 256 (1 - 2048)
Memory Regions: 65536 (1 - 524288)
Resource Groups: 2 (1 - 128)
Class of Service: 5

Receive Queue Count: 512 (1 - 1000)
Transmit Queue Count: 64 (1 - 1000)
Completion Queue Count: 576 (1 - 2000)

Figure 38. VMMQ configuration with Cisco IMC

Conclusion

The Cisco UCS VIC can provide considerable flexibility and optimized performance for applications running on Cisco UCS rack and blade servers. Note, though, that performance can vary, and that performance tuning settings can vary from system to system and from data center to data center. You may need several iterations of testing to find the right settings for a given configuration.

For More Information

Cisco Unified Computing System

Cisco UCS Manager Configuration Common Practices and Quick-Start Guide

Managing Cisco UCS C-Series Rack Servers (white paper)

Cisco Configuration Limits

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **https://www.cisco.com/go/offices**.

Printed in USA LE-50402-01 05/21